From Collapse to Improvement: Statistical Perspectives on the Evolutionary Dynamics of Iterative Training on Contaminated Sources

Soham Bakshi
Department of Statistics
University of Michigan
bakso@umich.edu

Sunrit Chakraborty
Department of Statistical Science
Duke University
sunrit.chakraborty@duke.edu

Abstract

The problem of model collapse has presented new challenges in iterative training of generative models, where such training with synthetic data leads to an overall degradation of performance. This paper looks at the problem from a statistical viewpoint, illustrating that one can actually hope for improvement when models are trained on data contaminated with synthetic samples, as long as there is some amount of fresh information from the true target distribution. In particular, we consider iterative training on samples sourced from a mixture of the true target and synthetic distributions. We analyze the entire iterative evolution in a next-token prediction language model, capturing how the interplay between the mixture weights and the sample size controls the overall long-term performance. With a non-trivial mixture weight of the true distribution, even if it decays over time, simply training the model in a contamination-agnostic manner with appropriate sample sizes can avoid collapse and even recover the true target distribution under certain conditions. Simulation studies support our findings and also show that such behavior is more general for other classes of models.

1 Introduction

Generative AI models, especially large language models (LLMs) based on transformers [46], such as ChatGPT [37, 38, 9], have revolutionized tasks like text generation, summarization, sentiment analysis, and text-to-image/audio translation. Other deep generative models such as diffusion models [28, 45], flow-based models [14, 15, 32], GANs [24, 36, 2], and VAEs [31] have similarly advanced data synthesis by learning complex distributions from massive datasets. However, these models are highly data-hungry; for example, GPT-3 has 175 billion parameters and was trained on 570 GB of text.

As high-quality, human-generated data becomes scarce [10, 4], models increasingly rely on synthetic data generated by earlier versions. With AI-generated content now pervasive online, e.g., LLM-generated content accounts for 5% of Wikipedia [8], which was then used in training ChatGPT-4 [12]—training datasets are becoming increasingly self-referential. Similar trends are seen in vision datasets [41]. This feedback loop of iterative training on synthetic data, whether intentional or not, departs from classical i.i.d. assumptions and raises critical questions about the long-term behavior and evolution of such models.

Contributions. We study when contamination-agnostic iterative training can exploit a continuing, possibly vanishing, influx of fresh data from the true distribution. Our contribution is an exact characterization of the risk dynamics in a setting that simultaneously allows an unlabeled mixture of real and synthetic data, data accumulation, iteration-varying sample sizes n_t (or m_t) and evolving

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

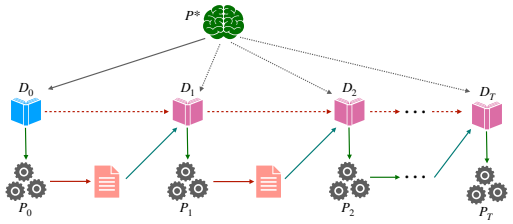


Figure 1: P^* is the true underlying data distribution, at time t, model M_t is trained on data D_t, where D_0 is purely a sample from P^*, but subsequently D_t consists of (i) synthetic part, (ii) accumulated part, (iii) fresh information. Can P_t approximate P^* well? In parametric setting, $P_t = P_{\hat{\theta}_t}$.

fresh-data proportions α_t, with training agnostic to the source of each sample. Specifically: (i) for a next-token prediction model we derive the exact finite-time risk recursion (Theorem 3.1) and identify joint conditions on (α_t, n_t) for collapse, iterative improvement and consistency, e.g., without accumulation, even $\alpha_t \downarrow 0$ with $\sum_t \alpha_t = \infty$ yields consistency when $n_t \asymp t^2$ (Proposition 3.4); (ii) we extend the exact analysis to accumulation combined with fresh mixture data (Section 3.2), showing, e.g., that a fixed number of new samples per round with a fixed $\alpha > 0$ already guarantees consistency; (iii) beyond the categorical model, we show that without fresh real data no estimator can improve on the initial fit, while with fresh data the population-level risk improves (Section 4); and (iv) simulations show that the predicted regimes persist for Gaussian, Gaussian mixture, logistic regression and GPT-2 models. The exact theory abstracts a language model to fixed-context categorical next-token distributions; representation learning, parameter sharing across contexts, optimization dynamics and sequence-level feedback are outside its scope. A detailed comparison with prior work is given in Appendix G.

The rest of the paper is organized as follows. In Section 2, we introduce our theoretical framework and review the existing literature. In Section 3, we analyze the iterative evolution in depth for the setting of next-token prediction language models. In Section 4, we generalize our insights for a single step of this iterative training by asking when it is possible to improve the estimator statistically. Section 5 provides extensive simulations to support the theoretical insights and demonstrate the generality of our results.

2 Theoretical Setup

We begin our discussion with a general mathematical formulation of this iterative training process. We illustrate the formulation using LLMs such as ChatGPT for natural language generation as an example. Assume that there is a fixed language model which governs the way humans generate text documents and treat this as the ground-truth data distribution. The goal of a language model (e.g. ChatGPT) is to learn this true data distribution through samples of text documents with the aim of generating from that distribution. When the first version of GPT was trained, the source of training data was mostly an online corpus consisting of predominantly samples from the true data distribution or human language model. However, while training ChatGPT3, the online corpus consisted of texts from different sources — some of it might already be present when training ChatGPT2, while the new training data (i.e., not used previously) could either be written by humans (*real* data) or generated by a previous version of GPT (*synthetic* data). However, during the training process, the source of each sample in the training data is unknown, and as such, the model is trained on the entire dataset in a manner agnostic to whether the data are real or synthetic. The question is whether such training improves the performance of the model over iterations statistically, in the sense of approximating the true human language model.

Mixture modeling: Mathematically, let P^* denote the true data distribution and $\mathcal{P} = \{P_\theta : \theta \in \Theta\}$ be the class of models. At time $t = 0, 1, \ldots,$ consider a data corpus $D_t \sim P_t$ of size $|D_t| = n_t$ based on which the model is trained and we denote the fitted model as $\hat{P}_t := P_{\hat{\theta}_t}$, where $\hat{\theta}_t$ is the estimator based on D_t obtained by optimizing some loss function (e.g., it could be L_2-loss for regression, or logistic loss or a more general negative log-likelihood loss). At the first stage $P_0 = P^*$, i.e. the data corpus D_0 consists of samples only from the true data distribution. Subsequently, at $t > 0$,

the training data D_t consists of samples from mainly two different source types (i) *fresh* samples from true data distribution P^* (which we refer to as *real data*), and (ii) *synthetic data* generated from previously fitted models $\{\hat{P}_s : s < t\}$. Another potential source is *reusing data from prior times* (accumulation), see Section 3.2; however, our focus is primarily on the two main sources. For simplicity, assuming that only the latest version of the model is used for the synthetic generation, then the data generative mechanism at the $t+1$ time point can be modeled as a mixture

$$P_{t+1} = \alpha_{t+1} P^* + (1 - \alpha_{t+1}) \hat{P}_t \tag{1}$$

where α_t captures the proportion of *fresh* real data in the t-th stage data D_t. In practice, the training data consists of an unlabeled mixture of samples from different sources, and the learner is agnostic to these sources. Our population level mixture formulation provides an idealized approximation to such training regimes, where the mixture proportions capture the effective contribution of real and synthetic data. The complete training process is illustrated in Figure 1. Note that even though the data arises from different heterogeneous sources, the model is still fitted using the same optimization method with the fixed model class $\mathcal{P}$. It is also important to recognize that the mixture assumption is at the population level – this is different from an empirical mixture involving adding some of the real samples from the first time point to potentially all synthetic samples generated at time t.

The goal of this study is to understand under what conditions the iterative training with a mixture of real and synthetic data can actually improve the models over time. Precisely, our interest is to examine how the expected distance of the fitted model from the true distribution, $\mathbb{E}d(P^*, \hat{P}_t)$ evolves with training iterations $t \geq 0$, where d is some appropriate distance measure. We specifically explore the conditions on the mixture weights $(\alpha_t)_{t\geq 1}$ and sample sizes $(n_t)_{t\geq 0}$ such that not only collapse is avoided, but also models eventually improve with training iterations, i.e. $\mathbb{E}d(P^*, \hat{P}_t) \lesssim \mathbb{E}d(P^*, \hat{P}_0)$ and ideally achieve consistency in the limit, i.e. $\mathbb{E}d(P^*, \hat{P}_t) \to 0$ with growing sample sizes $n_t \uparrow \infty$ as $t \to \infty$.

Related Work: The problem of model collapse was first identified and studied in [44] and was empirically validated by [1, 6, 34, 35, 26] for a wide variety of models and web-scale datasets. However, there are relatively fewer works that theoretically study the effect of model collapse. Theoretical analysis of model collapse was done in [43] for Gaussian data, [16] in regression settings and [18] studies changes in scaling laws when iterative training is done with synthetic data.

[1] conducts experiments across different mixed training setups, indicating that incorporating real data can help alleviate model collapse; we supply the corresponding exact risk trajectory (Theorem 3.1) and precise conditions on (α_t, n_t) for improvement and consistency. For simplified language models, [42] examine training with discrete mixtures of real and synthetic data, deriving upper bounds on synthetic data proportions to prevent collapse. However, note our population level mixture modeling differs from data mixtures considered in [42], where real data is injected at each training step from a fixed (finite) original sample. They ask how much synthetic data can be included while avoiding collapse; a fixed original sample provides no increasing information about P^*, so their analysis does not address improvement or consistency, which require a continuing arrival of fresh data (Section 4). From [42] we inherit only the categorical next-token abstraction; the evolving population-level mixture, the exact risk trajectory and the improvement/consistency results are new here. [5] considers a general maximum likelihood estimation framework in which the training data at each iteration combines the fixed original real sample D_0 with synthetic samples (no fresh real data), with the real and synthetic parts known to the learner, and analyze the stability of training process at the distribution level under a locality assumption in parameter space. For diffusion models, [21] provides a bound of the TV distance between $\hat{P}_t$ and P^*, and also shows the importance of balancing portions of real and synthetic data. Our work is in similar spirit with the above settings, but is directed towards studying statistical improvement or consistency instead of just focusing on preventing collapse. Furthermore, in contrast with most existing works, we consider asymptotic regimes with sample sizes growing at each iteration and seek to characterize the limiting behavior of model risk.

Finally, we note that several recent studies have emerged as potential solutions to model collapse. A line of work [23, 30] explores *data accumulation* to prevent model collapse, although accumulation alone cannot improve performance over the iterations (see Section 3.2); see also [13] for accumulation in general M-estimation and [3] for source-agnostic iterative MLE on accumulated data, both without fresh real data. While few recent works investigate strategies for curating or filtering synthetic data to prevent model collapse [19, 20], a thorough evaluation of their effectiveness remains absent. A

detailed comparison with existing work, including a table stating for each closely related work its data-generating setting, the quantity studied and what is new here, is given in Appendix G.

3 Iteratively Trained Language Models

To theoretically study the evolution of the iterative training process for large language models, we focus on a simplified statistical language model with context-based next-token prediction. This modeling approach is similar to [42]; further details and background on language models are provided in Appendix A. Fixing a context and K target tokens, let $\theta^* \in \Delta^{K-1}$ denote the true conditional probabilities of the next token, where $\Delta^{K-1} = \{x \in \mathbb{R}^K : x_i \geq 0, \sum_i x_i = 1\}$ is the $(K-1)$-dimensional probability simplex. Theoretically, this set-up is categorical proportion estimation with model class $\mathcal{P} = \{\text{Cat}(\theta) : \theta \in \Delta^{K-1}\}$, where Cat denotes a categorical distribution. Without parameter sharing, one-hot contexts decouple into independent categorical problems, so the aggregate risk over contexts is a context-frequency-weighted sum of per-context trajectories, each governed by the results below (Appendix A). Representation learning, parameter sharing across contexts, optimization dynamics and sequence-level feedback are outside the scope of this theory; the GPT-2 experiment in Section 5 checks whether the predicted regimes persist in their presence.

Iterative Training Process: Now, let us frame the problem for iterative training setting where the next stage new data is generated from a categorical distribution whose class probabilities are a mixture of the true class probabilities and the current fitted probabilities. The first stage data $Y^0 = \left(Y_1^0, \ldots, Y_{n_0}^0\right)$ are n_0 i.i.d. samples from the true data distribution $P_{\theta^*} = \text{Cat}(\theta^*)$, using them we get the initial estimates as $\widehat{\theta}_0(k) = \frac{1}{n_0}\sum_{j=1}^{n_0} \mathbf{1}[Y_j^0 = k]$. For any training time $t \geq 0$, given the current estimate $\widehat{\theta}_t$, the next stage data $Y^{t+1} = (Y_1^{t+1}, \ldots, Y_{n_{t+1}}^{t+1})$ is generated iid from

$$P_{t+1} := \text{Cat}(\theta_{t+1}) = \alpha_{t+1}P_{\theta^*} + (1-\alpha_{t+1})\widehat{P}_t = \text{Cat}(\alpha_{t+1}\theta^* + (1-\alpha_{t+1})\widehat{\theta}_t),$$

which recursively yields the next stage estimate $\widehat{\theta}_{t+1}$ using the corresponding sample proportions from Y^{t+1} (agnostic of data source). Our aim here is to understand the entire trajectory of $\hat{\theta}_t$ (equivalently $\widehat{P}_t = P_{\hat{\theta}_t}$) in terms of $R_t := \mathbb{E}\|\hat{\theta}_t - \theta^*\|_2^2$, the risk of t-th estimate, measuring the quality of model fit.

Theorem 3.1. *In the above setting, the sequence $(R_t)_{t\geq 1}$(as defined above) satisfies the following recurrence*

$$R_t = \frac{n_0}{n_t}R_0 + \frac{n_t - 1}{n_t}(1-\alpha_t)^2 R_{t-1}, \text{ where } R_0 = \frac{\sum_k \theta^*(k)(1-\theta^*(k))}{n_0} = \frac{1-\|\theta^*\|_2^2}{n_0}.$$

The recursion shows that the risk evolves as a combination of a variance term from fresh data and a propagated error term from previous iterations, modulated by the mixture weight. From now on, let R_∞ denote the limiting model risk $\lim_{t\to\infty} R_t$, when the limit exists (otherwise $\liminf_t R_t$). Observe that the trajectory of $(R_t)_{t\geq 1}$ is governed by the mixture weights and sample sizes. Under different mixing regimes, we study the long-term behavior of the sequence $(R_t)_{t\geq 0}$ based on the limiting risk $R_\infty := \lim_{t\to\infty} R_t$ (when it exists), relative to the initial risk R_0:

1. **Consistency:** models consistently improve to achieve vanishing risk in the limit, $R_\infty = 0$.
2. **Iterative Improvement:** models improve with time or stabilize at a risk level better than or close to the initial risk, $R_\infty \leq R_0$.
3. **Collapse:** models fail to improve, limiting risk exceeds the standard risk, e.g. $R_\infty > R_0$.

First we analyze the case when the mixture weight of real data is bounded away from 0.

Corollary 3.2. *Assume the mixture weight of real data maintains $\inf_t \alpha_t = \alpha > 0$. Then based on the evolution of the training size n_t, the following holds:*

1. *If $n_t = n$ (with $n_0 = n$), then $R_\infty \in [R_0, R_0(1 - \frac{n-1}{n}(1-\alpha)^2)^{-1}] \subseteq [R_0, nR_0)$.*
2. *If $n_t \uparrow \infty$, then $R_\infty = 0$ (consistency for any increasing sample size sequence n_t).*

Note with fixed sample size, future models can never be better than the initial model, but the gap $R_\infty - R_0$ depends on the mixture weights and is smaller for bigger α.

Remark 3.3. The proof of the above corollary demonstrates that if $\alpha_t = \alpha$ is fixed and $n_t = n$ for all t, then the limiting risk is *exactly* equal to the upper bound. The bound appears due to the condition that $\inf_t \alpha_t = \alpha$. When $\alpha = 0$ (purely synthetic data), the limit becomes nR_0, while when $\alpha = 1$ (fresh real data), the limit is R_0, as expected.

Now, let us consider the situation when the true mixture weights satisfy $\alpha_t > 0$ but decay to 0.

Proposition 3.4. *(Decaying Real-Data Mixture Weights)*

1. *For any sequence $\alpha_t \downarrow 0$, a 'large enough' sample size $n_t > \lceil \frac{n_{t-1}-1}{\alpha_t(2-\alpha_t)} \rceil + 1$ ensures t-th stage improvement i.e. $R_t < R_{t-1}$. Consequently, if the above recursive inequality is satisfied for all $(n_t : t \geq 1)$, then $(R_t)_{t\geq 0}$ is a strictly decreasing sequence and $R_\infty < R_0$.*

2. *If α_t satisfies $\sum_{t=1}^{\infty} \alpha_t = \infty$, then any sequence of sample sizes with $\sum_{t=1}^{\infty} 1/n_t < \infty$ ensures consistency, that is $R_\infty = 0$.*

Remark 3.5. The condition in Part (2) can be interpreted as *sufficiently slowly decaying* α_t and examples include $\alpha_t \asymp 1/\log(t)$ or $\alpha_t \asymp 1/\sqrt{t}$ or $\alpha_t \asymp 1/t$ or $\alpha_t \asymp 1/t(\log t)^\gamma$ for some $\gamma \in (0, 1)$ – in all such cases, the choice of $n_t \asymp t^2$ ensures consistency.

The above result provides sufficient (but not necessary) conditions under which improvement or consistency is achieved at the limit $t \to \infty$. Unlike Case 2 of Corollary 3.2, where any sequence $n_t \uparrow \infty$ ensures consistency, these conditions on the evolution of sample sizes depend on the sequence of decaying mixture weights. Observe (from part 1) that to ensure improvement n_t needs to grow faster for smaller weights α_t on the true distribution, essentially suggesting adjusted scaling laws while training with synthetic data, as shown in [18]. One potential limitation of Proposition 3.4 is that the sample size n_t depends on the potentially unknown α_t, which is problematic in real-life scenarios – although this is expected. However, the purpose of the result is to demonstrate that there exist sequences n_t such that estimation can indeed be improved even by agnostic (agnostic to the data source labels) model fitting with contaminated data.

Using a classifier before training

Recent work [47, 19] proposes filtering synthetic data using a classifier to mitigate model collapse. In our framework, let $\mathcal{C}_t$ distinguish between P^* and $\hat{P}_t$ with type I and II errors e_{1t}, e_{2t}. Given $Y^{t+1} \sim \alpha_{t+1}P^* + (1-\alpha_{t+1})\hat{P}_t$, we retain only samples classified as "real," yielding filtered data $\tilde{Y}^{t+1}$. Then $\tilde{Y}^{t+1}$ follows a mixture with updated weight

$$\tilde{\alpha}_{t+1} = \frac{\alpha_{t+1}(1-e_{1t})}{\alpha_{t+1}(1-e_{1t}) + (1-\alpha_{t+1})e_{2t}}, \quad \tilde{n}_{t+1} = \big[\alpha_{t+1}(1-e_{1t}) + (1-\alpha_{t+1})e_{2t}\big]n_{t+1}.$$

For a random classifier ($e_{1t} = e_{2t} = 1/2$), $\tilde{\alpha}_{t+1} = \alpha_{t+1}$ and $\tilde{n}_{t+1} = n_{t+1}/2$.

Oracle classifier. One can generate infinite samples from $\hat{P}_t$ (pre-trained model from last step). Assume that there is an oracle who also has access to infinite samples from P^* which attains the information-theoretic lower bound $e_{1t} + e_{2t} \geq 1 - \mathsf{TV}(P^*, \hat{P}_t)$ (e.g., the Neyman-Pearson Likelihood Ratio Test) – for a fixed type 1 error rate $e_{1t} = e_1$, it has a type 2 error rate $e_{2t} = 1 - e_1 - \mathsf{TV}(P^*, \hat{P}_t)$, yielding

$$\tilde{\alpha}_{t+1} = \frac{\alpha_{t+1}(1-e_1)}{1 - e_1 - \frac{1}{2}(1-\alpha_{t+1})\|\hat{\theta}_t - \theta^*\|_1} \;\geq\; \alpha_{t+1}.$$

Thus filtering effectively increases the contribution of real data. For a label-agnostic classifier with fixed error rates (e_{1t}, e_{2t}), filtering is an exact reweighting, and the risk follows the recursion of Theorem 3.1 with (α_{t+1}, n_{t+1}) replaced by $(\tilde{\alpha}_{t+1}, \tilde{n}_{t+1})$ (Theorem E.2). For adaptive classifiers whose errors depend on the current fit, such as the oracle classifier above, this reweighting holds per step, conditionally on $\hat{\theta}_t$, but we do not claim an exact closed-form trajectory. See Appendix E, F for details.

3.1 Iterative Retraining On Purely Synthetic Data

Now, we turn to the case when there is no fresh real data after D_0, i.e. for all $t \geq 1$, $\alpha_t = 0$ and the t-th model $\widehat{P}_t$ is trained on *purely* synthetic data Y^t, generated from previous model fit $\widehat{P}_{t-1}$. For fixed sample sizes $n_t = n$, collapse cannot be avoided and the limiting risk is nR_0, as discussed in Remark 3.3. The following corollary shows that increasing n_t might be beneficial in this case.

Corollary 3.6. *For iterative retraining with purely synthetic data with sample sizes $(n_t : t \geq 0)$, the following holds: $n_0 R_0 \sum_{s \leq t}(1/2^{t-s} n_s) < R_t < n_0 R_0 \sum_{s \leq t}(1/n_s)$.*

Unlike in the previous settings, there is no hope of improvement in the sense $R_t < R_0$ by growing sample size n_t. However, R_t can be made arbitrarily close to R_0 (this is intuitively obvious since if we have access to an unlimited number of samples from P_{θ_t}, then θ_t can be estimated without any statistical error, hence we only incur error at the first round with finite n_0 samples from P^*).

3.2 Effect of data accumulation

To mitigate model collapse, a line of work [23, 30] suggests training with accumulated data from past stages. Here, we study the evolution of iterative training in a setting involving both data accumulation and fresh samples drawn from a mixture distribution. At iteration $t = 0$, we obtain $\widehat{\theta}_0$ based on n_0 i.i.d. samples from the true data distribution P_{θ^*}. Subsequently, at iteration $t \geq 1$, $\widehat{\theta}_t$ is obtained by training the model on dataset D_t of size $n_t = n_{t-1} + m_t$, with $D_t = D_{t-1} \cup \tilde{D}_t$ containing the n_{t-1} samples in D_{t-1} that were used to train the model at iteration $t-1$ (accumulation part), and m_t new (previously unseen) samples generated conditionally i.i.d. (given $\hat{\theta}_{t-1}$) from

$$Q_t = \alpha_t P_{\theta^*} + (1 - \alpha_t)\widehat{P}_{t-1}.$$

Previous studies involving accumulation consider the case when the new m_t samples are purely synthetic, which is only a special case in our general setting corresponding to $\alpha_t = 0$. Denote $R_t = \mathbb{E}\|\hat{\theta}_t - \theta^*\|_2^2$. We assume that we use MLE on the entire training dataset to obtain $\hat{\theta}_t$, which implies that the model training is agnostic to the nature of the source of the particular samples. As we show in Appendix B.1, in this setting, the risk satisfies the following recurrence

$$R_t = \left(\gamma_t^2 - \frac{m_t(1-\alpha_t)^2}{n_t^2}\right) R_{t-1} + \frac{m_t n_0}{n_t^2} R_0,$$

where $\gamma_t = 1 - \frac{\alpha_t m_t}{n_t} \in (0, 1]$. Note that without accumulation, setting $n_t = m_t$, we recover the recursion from Theorem 3.1. As before, we analyze the asymptotic properties of $\{R_t\}$ in different special cases and key findings are summarized in the following proposition.

Proposition 3.7. *For the setting considered above, the following special cases hold.*

1. ***Accumulation with purely synthetic case*** *($\alpha_t = 0$): In this case, R_t is monotonically increasing and we have $R_0 < R_t < n_0 R_0$ for any $n_0 > 1$. Additionally, if the new data per iteration has fixed size $m_t = m$, then the upper bound can be improved to $R_t \leq (2 - 1/n_0)R_0 - (n_0 - 1)R_0/n_t$ for any m. For $m = n_0$, the upper bound becomes $R_t \leq R_0(1 + (1 - 1/n_0)(\pi^2/6 - 1))$. Thus, in both these cases, asymptotically $R_\infty < 2R_0$.*
2. ***Linear Data Growth*** *($m_t = m$ with $\alpha_t > 0$): Consider the two sub-cases.*
   - *Fixed $\alpha_t = \alpha > 0$: In this case, $R_t \to 0$ and the rate is controlled by α.*
   - *Decaying $\alpha_t \sim ct^{-\beta}$: In this case, $R_\infty > 0$ (no consistency). If $\beta > 1$, then $R_\infty > R_0$. If $\beta < 1$, then $R_\infty < R_0$ (iterative improvement). For $\beta = 1$, it depends on c, n_0, m.*
3. ***Geometric Data Growth*** *($m_t = c_0 n_{t-1}$ with $\alpha_t > 0$):*
   - *Fixed $\alpha_t = \alpha > 0$: Here $R_t \to 0$.*
   - *Decaying $\alpha_t \sim ct^{-\beta}$: If $\beta \leq 1$, then $R_t \to 0$. If $\beta > 1$, then $R_\infty > 0$. Again, whether iterative improvement occurs depends on c_0, n_0, c and β.*

Our findings (part 1 of the above proposition) align with the results in the previous works on accumulation [23, 30]. For fixed sample sizes, Proposition 3.7 (part 1) gives a stronger upper-bound

on the limiting risk than what we would obtain by standard training, demonstrating the benefit of accumulation. However, without fresh real data, R_t is strictly increasing and although the limiting risk is improved (compared to the setting without accumulation), there is no improvement of the estimator beyond the first step.

4 Importance of fresh real data to improve iterative training

Results from previous sections show (i) with fresh real data at each stage, improvement is always possible and consistency also holds in many cases, (ii) with no fresh real data after D_0, it is not possible (even with accumulation) to do better than the initial fit. In this section, we argue that this is a more general phenomenon. Consider a *single step* of the general iterative training process described in Section 2, with model class $\mathcal{P} = \{P_\theta : \theta \in \Theta\}$. Let $\hat{\theta}_0$ be an estimator trained using data D_0 and $\hat{\theta}_1$ be the estimator based on D_1 in the following iteration. Our aim is to understand under what conditions $\hat{\theta}_1$ can be statistically better than $\hat{\theta}_0$. The following two results respectively deal with the necessity and sufficiency of fresh samples from the true data distribution to improve estimation. Denote $\mathcal{R}(\theta, P) := \mathbb{E}[\ell(\theta; Z)]$ where ℓ is some loss function, and the expectation is taken with respect to $Z \sim P$ and any randomness in θ (if estimator). Typical examples with $Z = (X, Y)$ include the regression setting with $\ell(\theta; Z) = (Y - X^\top\theta)^2$ (linear regression), $\ell(\theta, Z) = (Y - X^\top\theta)^2 + \lambda \|\theta\|_1$ (LASSO regression), classification setting with $\ell(\theta, Z) = -Y \log(\sigma(X^\top\theta)) - (1-Y)\log(1-\sigma(X^\top\theta))$ (for logistic regression), $\ell(\theta, Z) = \max\{0, 1 - YX^\top\theta\}$ (for linear SVM) and with $Z = X$, the density estimation setting with $\ell(\theta, X) = -\log p_\theta(X)$.

Theorem 4.1. *If $D_1|D_0$ is independent of P^* and known, then for any convex loss $\ell(\theta; X)$ (convex in θ), for any estimator $\hat{\theta}_1$ based on D_1, there exists $\hat{\theta}_0$ based on D_0, such that $\mathcal{R}(\hat{\theta}_0, P^*) \le \mathcal{R}(\hat{\theta}_1, P^*)$.*

The result shows that if D_1 has no additional information about P^* other than that already in D_0, then for any estimator based on D_1, one can construct an estimator based only on D_0 which outperforms the former. It is important that the conditional distribution of D_1 given D_0 is known; otherwise, although $\hat{\theta}_0 = \mathbb{E}[\hat{\theta}_1|D_0]$ has better risk, it is not a computable statistic from the data. As an example, suppose given D_0, an adversary chooses a distribution for D_1 without any knowledge of P^* – in this case, the learner without access to this extra information cannot compute $\mathbb{E}[\hat{\theta}_1|D_0]$. Secondly, the result does not say that given an estimation method (e.g. empirical risk minimization), using it on D_0 is as good as using it on D_1. In the mixture setting $P_1 = \alpha P^* + (1-\alpha)\hat{P}_0$, where $\hat{P}_0$ is a function of D_0; hence for $\alpha = 0$, $D_1|D_0$ is independent of P^*. It is interesting to note that the result applies even to the case of accumulation, since $D_1|D_0$ is still independent of P^* (reusing 'real' data from D_0 is very different from having 'fresh real' data).

We next present a partial converse to the above result, demonstrating that availability of fresh information injected into the training actually helps the statistical estimation. In this, we consider the mixture model setting discussed in the previous section. In this case, since we only consider a single step of the overall iterative evolution, consider an arbitrary $\hat{\theta}_0$. In the next step, under the mixture model, the underlying population distribution is $Q_1 = \alpha P^* + (1-\alpha)P_{\hat{\theta}_0}$ – see Figure 5 for a visualization. Consider the negative log likelihood loss $\ell(\theta, X) = -\log p_\theta(X)$. Minimizing the risk with respect to this loss at the empirical level is equivalent to MLE, which at the population level is equivalent to the KL-projection of Q_1 on the model space $\mathcal{P}$. The next result shows that at the population level, the risk can be improved if $\alpha > 0$.

Theorem 4.2. *For $\ell(\theta, X) = -\log p_\theta(X)$, given an arbitrary $\theta_0 \in \Theta$, $Q_1 = \alpha P^* + (1-\alpha)P_{\theta_0}$ and $\theta_1 = \arg\min_\theta \mathcal{R}(\theta, Q_1)$, we have $\mathcal{R}(\theta_1, P^*) \le \mathcal{R}(\theta_0, P^*)$ if $\alpha > 0$.*

It is worth emphasizing that the result is at a population level – for sample-based estimators, the statistical error due to finite sample size also enters the picture. However, it shows that with sufficiently large sample size n, the estimator has better risk. Lastly, if $\mathcal{P}$ is convex (closed under mixtures), θ_0 does not minimize $\mathcal{R}(\cdot, P^*)$, $P_{\theta^*} \ll P_{\theta_0}$ for $\theta^* := \arg\min_\theta \mathcal{R}(\theta, P^*)$, and the relevant risks are finite, then the inequality is strict (Proposition C.1 in Appendix C).

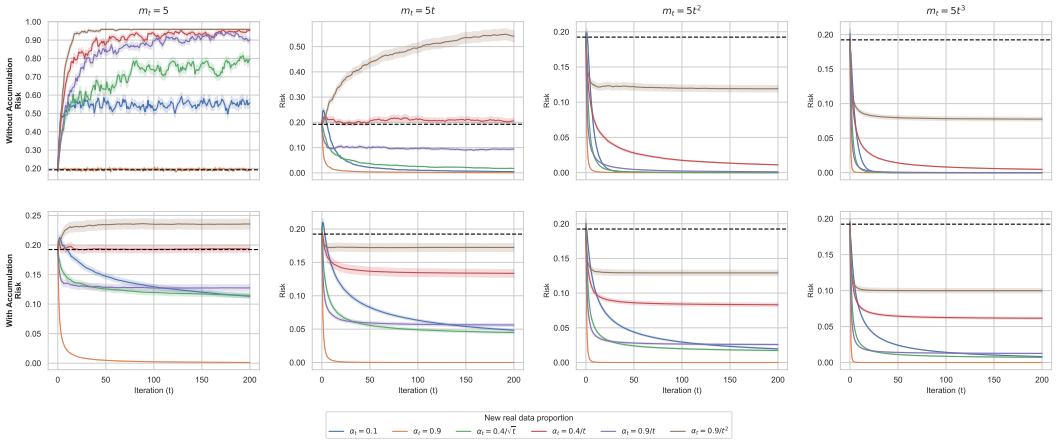


Figure 2: Iterative evolution of estimation quality for multinomial model under various settings corresponding to m_t (size of new samples at iteration t) and α_t (fresh data proportion among new samples) – top row shows the results without accumulation, bottom row shows results with accumulation. Each column corresponds to a different data growth regime: note that the total size of the data used in iteration t is $n_t = m_t$ (without accumulation) and $n_t = n_{t-1} + m_t$ (with accumulation). The black dashed horizontal line is the value of $R_0 = (1 - \|\theta^*\|_2^2)/5$, as $m_0 = 5$.

5 Numerical experiments

We complement our analysis with several numerical experiments. The purpose of the experiments is to understand the iterative evolution of the estimation performance R_t for different types of models under different rates for the proportion of ground truth α_t, and sample size n_t. We present the results for the multinomial model, which support our results and also shed light on related heuristics. Due to space constraints, we postpone additional results for this model, along with results for the Gaussian model, Gaussian mixture model and logistic regression model to Appendix D.2. The results hint at the wider applicability of the results presented in this work. Finally, we present a numerical experiment using the nano-GPT model, simulating a realistic evolution of training under synthetic data.

Multinomial model We study the case of iterative training of multinomial models when the training data at each iteration is corrupted by synthetic samples generated from the trained model at the previous iteration. We choose $K = 50$, $T = 200$ and perform experiments under different combinations of the proportion of ground-truth α_t and sample sizes m_t (see notations in Section 3.2). For each experiment, at iteration $t = 0$, the model is trained on m_0 samples from the true distribution P_{θ^*} and at $t \geq 1$, new data of size m_t is drawn from the mixture $\alpha_t P_{\theta^*} + (1 - \alpha_t) P_{\hat{\theta}_{t-1}}$, and then the model is trained (using MLE without any additional mixture-based modification) on either (i) only the m_t new samples (without accumulation), or (ii) combined data containing the m_t new samples and n_{t-1} samples used at iteration $t - 1$. Thus, the size of the training data at iteration t is $n_t = m_t$ (without accumulation) or $n_t = n_{t-1} + m_t = \sum_{s \leq t} m_s$ (with accumulation). For each of the different settings, we iteratively train the model and compute the risk $\hat{R}_t = \|\hat{\theta}_t - \theta^*\|_2^2$, $t = 1, \ldots, T$. Each experiment is repeated 100 times and the mean is shown, along with the standard error in Figure 2. The columns correspond to different rates of data growth m_t. Within each plot, different regimes based on α_t (proportion of fresh data) are shown, and these regimes are described in the corresponding legend of the plot. Figure 2 first shows that with α_t fixed or decaying slowly, it is possible to recover the true parameter, by training the model on corrupted data in an agnostic fashion, supporting many of the conclusions of Proposition 3.4. Secondly, for rapidly decaying α_t, although we do not see consistency, for fast $m_t \succ t^3$, we see that $R_\infty < R_0$, demonstrating that iterative training actually improves estimation. Finally, a direct comparison between training with and without accumulation shows that accumulation improves the performance when m_t is fixed, but for increasing m_t, this is not exactly true – for example, for $\alpha_t \asymp 1/t$, neither the choice of $m_t \asymp t^2$ or t^3 leads to consistency.

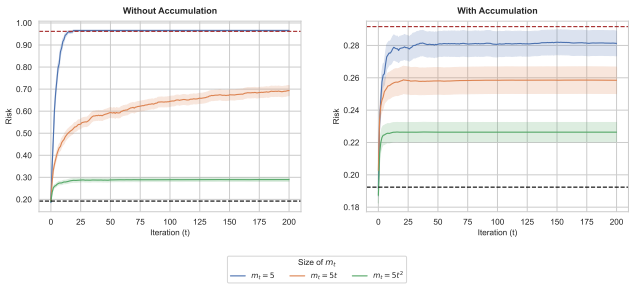


Figure 3: Iterative evolution of estimation quality for multinomial model under purely synthetic regime ($\alpha_t = 0$), when models are trained without accumulation (left) and with accumulation (right).

We also perform a simulation study to understand iterative training with a purely synthetic data regime (corresponding to $\alpha_t = 0$). The result is shown in Figure 3, for different choices of m_t, with the two plots corresponding to with (right) and without (left) data accumulation. The results in the right plot match our findings in Proposition 3.7 (part 1): we see that R_t increases with t till it stabilizes. A faster rate of growth for n_t simply reduces this limit, but the performance is always worse than R_0. This can be seen as a specific case of our general results in Section 4, illustrating that unless fresh information from the ground-truth distribution is available, the estimation quality will only worsen over iterations. The brown dotted horizontal line at the top corresponds to $R_0(1 + (1 - 1/m_0)(\pi^2/6 - 1))$, the theoretical upper bound for risk in the constant m_t case. On the other hand, the results in the left plot match our results in Corollary 3.2 and Corollary 3.6, with $R_t \to m_0 R_0$ (brown horizontal dotted line at the top) in the constant m_t case, but R_t being bounded in the $m_t \asymp t^2$ case, since $\sum_t 1/t^2 < \infty$.

GPT-2 learning on real text data. In this section, we explore a more complex transformer-based language model and demonstrate its evolution under different iterative training settings. For these experiments, we use the Wikipedia movie plots dataset from Kaggle[1]. It consists of movie plots (we call these documents) of around 34000 movies. For each experiment, we hold out 2000 documents and all training is based on observations from the remaining documents and synthetically generated documents. For each setting, recursive training is done for $T = 10$ rounds, evaluating the fitted model based on perplexity on the held-out data corpus.

We employ a moderate-sized GPT-2 language model using the nanoGPT implementation by Andrej Karpathy [29] (additional details provided in Appendix D.3). The following settings are considered (with $n_0 = 2000$ and $a = 200$), where in each setting, at $t = 0$, the model is trained using n_0 real documents: (1) *Synthetic only (Setting 0)*: At $t \geq 1$, the model is trained entirely on $n_t = n_0 + at$ synthetic documents generated by the last-fitted model. (2) *Accumulation only (Setting 1)*: For $t \geq 1$, retain the training data from previous time point D_{t-1} and combine it with a new synthetic documents sampled from last fitted model, yielding $n_t = n_0 + at$. (3) *Fixed α (Setting 2)*: For $t \geq 1$, construct a training set of size $n_t = n_0 + at$ using $n_t/4$ synthetic documents generated from the last-fitted model and $3n_t/4$ fresh, previously unused real documents. This maintains a constant $\alpha_t = 0.75$ at each step. (4) *Decaying α (Setting 3):* For $t \geq 1$, combine at synthetic documents generated from the last-fitted model with n_0 fresh previously unused real documents. The training data size grows as $n_t = n_0 + at$, while the proportion of fresh documents decays as $\alpha_t = n_0/n_t$.

The findings of our simulation studies are given in Figure 4 – each colored plot indicates a particular setting, and we plot the average perplexity over 10 random repetitions of the experiment, with bands showing the standard error. Similar to our findings from the previous section (and the results in Appendix D.2), setting 0 leads to catastrophic collapse, as can be seen by the increase in perplexity. The other settings can be seen more clearly in the right plot in Figure 4. For both settings 1 and 3, we see that the model iteratively does not improve; however, it does not deteriorate too much. Lastly, we see that in setting 2, the model actually improves over iterations, shown by the decrease in perplexity.

[1] `www.kaggle.com/datasets/jrobischon/wikipedia-movie-plots`

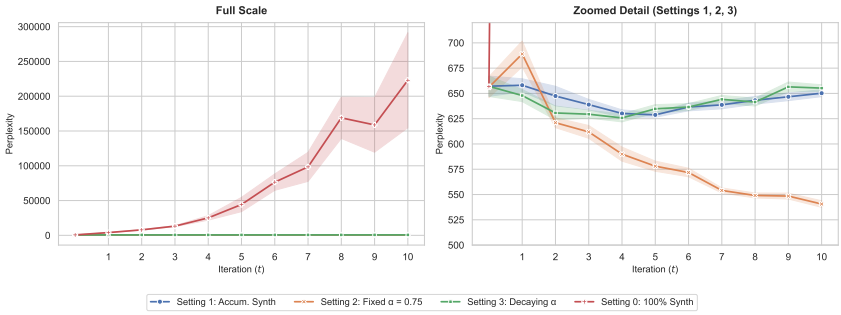


Figure 4: Iterative evolution of GPT-2-like language model evaluated based on perplexity on a held-out data corpus in 4 different settings, shown in the shared legend. Lines show the mean over 10 repetitions; shaded bands show the standard error.

6 Conclusion & Future Directions

We investigate model collapse in generative models and identify conditions under which statistical estimation can improve during iterative training with mixed real and synthetic data. In a simplified next-token prediction setting, we precisely characterize error evolution and show that consistency and improved estimation are achievable under suitable data mixing. Simulations suggest these findings generalize beyond the toy setting. The exact theory does not cover representation learning, parameter sharing across contexts, optimization dynamics or sequence-level feedback; extending it to these is an important direction. Crucially, we show that without continual infusion of real data, performance cannot improve, making retraining ineffective. Our analysis highlights how the balance between real data fraction and sample size governs long-term performance, offering testable predictions for practical applications.

While sample sizes can often be tracked, estimating the mixture weights remains a challenge. Leveraging recent advances in synthetic data detection (e.g., [39]), a crucial future direction is to estimate mixture proportions, which can then be used to obtain corrected estimate $\widehat{\theta}_t + \widehat{\alpha}_{t+1}^{-1}(\widehat{\theta}_{t+1} - \widehat{\theta}_t)$, potentially improving estimation. Although we do not directly control the proportion of synthetic data at each iteration, it is reasonable to expect lower synthetic contamination when the model is less accurate. In practice, when training language models on web-scale corpora, implicit human or algorithmic filtering naturally suppresses poor generations. This implies that the mixture weight $1 - \alpha_t$, reflecting the prior model's contribution, decreases with increasing risk R_{t-1}. Such *naturally adaptive control* reduces the influence of weak models, preventing collapse and keeping the limiting risk (R_∞) near or below the initial risk R_0, without requiring assumptions on n_t (see Appendix F).

While our framework models fresh information via a mixture distribution, there are other ways to get new information – most notably, reinforcement learning from human feedback (RLHF) to fine-tune LLMs or curate synthetic data [19, 20]. Our iterative setup also connects to information-passing scenarios involving limited fresh input, e.g., communicating LLMs or echo chambers [11]. In the future, it would be interesting to extend our theoretical framework to study a broader range of interactive training dynamics. Finally, assuming a fixed true data distribution may be unrealistic in evolving language environments. As AI-generated content fills up online text sources, the output from LLMs might also influence the language model of humans, i.e., the true data distribution. This feedback loop suggests a co-evolution of human and machine language, raising important questions about long-term model dynamics.

Acknowledgments and Disclosure of Funding

The authors are grateful to Ambuj Tewari for insightful discussions and helpful suggestions. This paper grew out of a project that was initiated in his course.

We received no third-party funding for this work and declare no competing interests.

References

[1] Sina Alemohammad, Josue Casco-Rodriguez, Lorenzo Luzi, Ahmed Imtiaz Humayun, Hossein Babaei, Daniel LeJeune, Ali Siahkoohi, and Richard G Baraniuk. Self-consuming generative models go mad. *arXiv preprint arXiv:2307.01850*, 2023.

[2] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *International conference on machine learning*, pages 214–223. PMLR, 2017.

[3] Daniel Barzilai and Ohad Shamir. When models don't collapse: On the consistency of iterative mle. *Advances in neural information processing systems*, 38:76813–76854, 2026.

[4] EM Bender and Batya Friedman. Data statements for nlp: toward mitigating system bias and enabling better science. In *Preprint at https://openreview. net/forum*, 2019.

[5] Quentin Bertrand, Avishek Joey Bose, Alexandre Duplessis, Marco Jiralerspong, and Gauthier Gidel. On the stability of iterative retraining of generative models on their own data. *ArXiv*, abs/2310.00429, 2023. URL `https://api.semanticscholar.org/CorpusID:263334017`.

[6] Matyas Bohacek and Hany Farid. Nepotistically trained generative-ai models collapse. *arXiv preprint arXiv:2311.12202*, 2023.

[7] Martin Briesch, Dominik Sobania, and Franz Rothlauf. Large language models suffer from their own output: An analysis of the self-consuming training loop. *arXiv preprint arXiv:2311.16822*, 2023.

[8] Creston Brooks, Samuel Eggert, and Denis Peskoff. The rise of ai-generated content in wikipedia. *arXiv preprint arXiv:2410.08044*, 2024.

[9] Tom B Brown. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020.

[10] Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, et al. A survey on evaluation of large language models. *ACM Transactions on Intelligent Systems and Technology*, 15(3):1–45, 2024.

[11] Matteo Cinelli, Gianmarco De Francisci Morales, Alessandro Galeazzi, Walter Quattrociocchi, and Michele Starnini. The echo chamber effect on social media. *Proceedings of the National Academy of Sciences*, 118(9):e2023301118, 2021.

[12] Maria del Rio-Chanona, Nadzeya Laurentsyeva, and Johannes Wachs. Are large language models a threat to digital public goods? evidence from activity on stack overflow. *arXiv preprint arXiv:2307.07367*, 2023.

[13] Apratim Dey and David Donoho. Universality of the $\pi^2/6$ pathway in avoiding model collapse. *arXiv preprint arXiv:2410.22812*, 2024.

[14] Laurent Dinh, David Krueger, and Yoshua Bengio. Nice: Non-linear independent components estimation. *arXiv preprint arXiv:1410.8516*, 2014.

[15] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp. *arXiv preprint arXiv:1605.08803*, 2016.

[16] Elvis Dohmatob, Yunzhen Feng, and Julia Kempe. Model collapse demystified: The case of regression. *arXiv preprint arXiv:2402.07712*, 2024.

[17] Elvis Dohmatob, Yunzhen Feng, Arjun Subramonian, and Julia Kempe. Strong model collapse, 2024. URL `https://arxiv.org/abs/2410.04840`.

[18] Elvis Dohmatob, Yunzhen Feng, Pu Yang, Francois Charton, and Julia Kempe. A tale of tails: Model collapse as a change of scaling laws, 2024. URL `https://arxiv.org/abs/2402.07043`.

[19] Yunzhen Feng, Elvis Dohmatob, Pu Yang, Francois Charton, and Julia Kempe. Beyond model collapse: Scaling up with synthesized data requires reinforcement, 2024. URL `https://arxiv.org/abs/2406.07515`.

[20] Damien Ferbach, Quentin Bertrand, Avishek Joey Bose, and Gauthier Gidel. Self-consuming generative models with curated data provably optimize human preferences, 2024. URL https://arxiv.org/abs/2407.09499.

[21] Shi Fu, Sen Zhang, Yingjie Wang, Xinmei Tian, and Dacheng Tao. Towards theoretical understandings of self-consuming generative models, 2024. URL https://arxiv.org/abs/2402.11778.

[22] Shi Fu, Yingjie Wang, Yuzhu Chen, Xinmei Tian, and Dacheng Tao. A theoretical perspective: How to prevent model collapse in self-consuming training loops. In *International Conference on Learning Representations*, volume 2025, pages 93254–93287, 2025.

[23] Matthias Gerstgrasser, Rylan Schaeffer, Apratim Dey, Rafael Rafailov, Henry Sleight, John Hughes, Tomasz Korbak, Rajashree Agrawal, Dhruv Pai, Andrey Gromov, et al. Is model collapse inevitable? breaking the curse of recursion by accumulating real and synthetic data. *arXiv preprint arXiv:2404.01413*, 2024.

[24] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

[25] Yanzhu Guo, Guokan Shang, Michalis Vazirgiannis, and Chloé Clavel. The curious decline of linguistic diversity: Training language models on synthetic text. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 3589–3604, 2024.

[26] Yanzhu Guo, Guokan Shang, Michalis Vazirgiannis, and Chloé Clavel. The curious decline of linguistic diversity: Training language models on synthetic text, 2024. URL https://arxiv.org/abs/2311.09807.

[27] Ryuichiro Hataya, Han Bao, and Hiromi Arai. Will large-scale generative models corrupt future datasets? In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 20498–20508. IEEE, 2023.

[28] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

[29] Andrej Karpathy. NanoGPT. https://github.com/karpathy/nanoGPT, 2022.

[30] Joshua Kazdan, Rylan Schaeffer, Apratim Dey, Matthias Gerstgrasser, Rafael Rafailov, David L Donoho, and Sanmi Koyejo. Collapse or thrive? perils and promises of synthetic data in a self-generating world. *arXiv preprint arXiv:2410.16713*, 2024.

[31] Diederik P Kingma. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[32] Durk P Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. *Advances in neural information processing systems*, 31, 2018.

[33] Matteo Marchi, Stefano Soatto, Pratik Chaudhari, and Paulo Tabuada. Heat death of generative models in closed-loop learning. In *2024 IEEE 63rd Conference on Decision and Control (CDC)*, pages 1524–1530. IEEE, 2024.

[34] Gonzalo Martínez, Lauren Watson, Pedro Reviriego, José Alberto Hernández, Marc Juarez, and Rik Sarkar. Combining generative artificial intelligence (ai) and the internet: Heading towards evolution or degradation? *arXiv preprint arXiv:2303.01255*, 2023.

[35] Gonzalo Martínez, Lauren Watson, Pedro Reviriego, José Alberto Hernández, Marc Juarez, and Rik Sarkar. Towards understanding the interplay of generative artificial intelligence and the internet. In *International Workshop on Epistemic Uncertainty in Artificial Intelligence*, pages 59–73. Springer, 2023.

[36] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.

[37] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. 2018.

[38] 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.

[39] Tara Radvand, Mojtaba Abdolmaleki, Mohamed Mostagir, and Ambuj Tewari. Zero-shot statistical tests for llm-generated text detection using finite sample concentration inequalities, 2025. URL `https://arxiv.org/abs/2501.02406`.

[40] Rylan Schaeffer, Joshua Kazdan, Alvan Caleb Arulandu, and Sanmi Koyejo. Position: Model collapse does not mean what you think. *arXiv preprint arXiv:2503.03150*, 2025.

[41] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in Neural Information Processing Systems*, 35:25278–25294, 2022.

[42] Mohamed El Amine Seddik, Suei-Wen Chen, Soufiane Hayou, Pierre Youssef, and Merouane Debbah. How bad is training on synthetic data? a statistical analysis of language model collapse, 2024. URL `https://arxiv.org/abs/2404.05090`.

[43] Ilia Shumailov, Zakhar Shumaylov, Yiren Zhao, Yarin Gal, Nicolas Papernot, and Ross Anderson. The curse of recursion: Training on generated data makes models forget. *arXiv preprint arXiv:2305.17493*, 2023.

[44] Ilia Shumailov, Zakhar Shumaylov, Yiren Zhao, Nicolas Papernot, Ross Anderson, and Yarin Gal. Ai models collapse when trained on recursively generated data. *Nature*, 631(8022): 755–759, 2024.

[45] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

[46] A Vaswani. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017.

[47] Jinghui Zhang, Dandan Qiao, Mochen Yang, and Qiang Wei. Regurgitative training: The value of real data in training large language models. *arXiv preprint arXiv:2407.12835*, 2024.

A Background on Language Models

Say we have a vocabulary of size K and $C \leq K^l$ many possible contexts of maximum length l. Assuming that the language data is generated from some unknown conditional probabilities given the contexts, the probability of the next token being $k \in [K] := \{1, \ldots, K\}$ given some context $j = (j_1, \ldots, j_\ell) \in [C]$ is denoted by $\mathbb{P}[Y = k \mid X = j]$ where X and Y are discrete random variables specifying a context and the next token respectively. In practice, conditional probabilities are unknown but we have access to a (large) corpus of training dataset $\{(x_l, y_l)\}_{l \in [N]}$ of N samples of contexts and next-token pairs represented by $x_l \in \{e_1, \ldots, e_C\}$ and $y_l \in \{e_1, \ldots, e_K\}$ where e_i's denote the canonical vectors. Then we consider estimating the conditional probabilities via the Softmax classifier, which minimizes the categorical cross-entropy loss function:

$$\underset{\mathbf{W}=[w_1,\ldots,w_K] \in \mathbb{R}^{C \times K}}{\arg\min} \; - \frac{1}{N} \sum_{l=1}^{N} y_l^\top \log \sigma\left(\mathbf{W}^\top x_l\right)$$

where $\sigma(v) = \frac{\exp(v)}{\sum_{k=1}^{K} \exp(v_k)}$ is the Softmax function and the functions exp and log are applied entry-wise. Note that we choose to work with one-hot embeddings as representations for tractable theoretical analysis even though current state-of-the-art language models, the x_l's are context representations computed using attention or transformer based networks. The estimates of conditional probability solving the Softmax problem are the empirical means:

$$\widehat{\mathbb{P}}[Y = k \mid X = j] = \frac{\exp\left(\hat{w}_k^\top e_j\right)}{\sum_{i=1}^{K} \exp\left(\hat{w}_i^\top e_j\right)} = \frac{1}{|\mathcal{C}_j|} \sum_{l \in \mathcal{C}_j} y_{lk}, \quad \text{with} \quad \mathcal{C}_j = \{l \in [N] \mid x_l = e_j\}.$$

Fixing a context say $j \in [C]$ (equivalently setting $x = e_j$), denote true conditional probabilities as

$$\theta^*(k) = \mathbb{P}\left[Y = k \mid X = j\right], \ \forall k \in [K], \quad \theta^* = [\theta^*(1) \cdots \theta^*(K)]^\top \in \Delta^{K-1}.$$

Then the above classification problem boils down to a statistical problem of categorical proportion estimation with data $\{(e_j, y_l)\}_{l \in \mathcal{C}_j}$. Here, the true data distribution $P_{\theta^*} := \text{Cat}(\theta^*)$ lies in the model class $\{\text{Cat}(\theta) : \theta \in \Delta^{K-1}\}$.

Aggregation over contexts. Since the one-hot Softmax problem has no parameter sharing, the estimates for different contexts are computed from disjoint subsets of the data and the iterative process decouples across contexts. Let π_j denote the frequency of context $j \in [C]$, $\theta^{*(j)}$ and $\hat{\theta}_t^{(j)}$ the true and fitted next-token probabilities for context j, $n_t^{(j)}$ the number of training samples with context j at iteration t, and $\alpha_t^{(j)}$ the corresponding fresh-data proportion. Treating the per-context sample sizes as given (equivalently, conditioning on them), the aggregate risk is

$$R_t^{\text{agg}} := \sum_{j=1}^{C} \pi_j \, \mathbb{E}\big\|\hat{\theta}_t^{(j)} - \theta^{*(j)}\big\|_2^2 = \sum_{j=1}^{C} \pi_j R_t^{(j)},$$

where each $R_t^{(j)}$ satisfies the recursion of Theorem 3.1 (or Theorem B.1 with accumulation) with $(n_t^{(j)}, \alpha_t^{(j)})$ in place of (n_t, α_t). Hence the results of Section 3 lift termwise: for instance, if every context with $\pi_j > 0$ satisfies the conditions for consistency, then $R_t^{\text{agg}} \to 0$, and if every such context satisfies $R_\infty^{(j)} < R_0^{(j)}$, then $R_\infty^{\text{agg}} < R_0^{\text{agg}}$.

B Proofs & Additional Results in Section 3

Proof of Theorem 3.1. For $t \geq 0$, firstly $Y^t = \left(Y_1^t, \ldots, Y_{n_t}^t\right)$ is iid data from $\text{Cat}(\theta_t)$ is used to make estimates

$$\widehat{\theta}_t(k) = \frac{\sum_{j=1}^{n_t} \mathbf{1}[Y_j^t = k]}{n_t}.$$

In first stage, $n_0\widehat{\theta}_0(k) \sim \text{Binom}(n_0, \theta^*(k))$ thus $\mathbb{E}\widehat{\theta}_0(k) = \theta^*(k)$ and $\mathsf{Var}\widehat{\theta}_0(k) = \frac{\theta^*(k)(1-\theta^*(k))}{n_0}$. Now for $t \geq 0$,

$$\widehat{\theta}_{t+1}(k) = \frac{\sum_{j=1}^{n_{t+1}} \mathbf{1}[Y_j^{t+1} = k]}{n_{t+1}},$$

and $\mathbf{1}[Y_j^{t+1} = k] \mid \widehat{\theta}_t \sim \text{Ber}(\alpha_{t+1}\theta^*(k) + (1-\alpha_{t+1})\widehat{\theta}_t(k))$, thus

$$\widehat{\theta}_{t+1}(k) \mid \widehat{\theta}_t \sim \frac{\text{Binom}(n_{t+1}, \alpha_{t+1}\theta^*(k) + (1-\alpha_{t+1})\widehat{\theta}_t(k))}{n_{t+1}}.$$

Then

$$\begin{aligned}
&\mathbb{E}[\widehat{\theta}_{t+1} \mid \widehat{\theta}_t] = \alpha_{t+1}\theta^* + (1-\alpha_{t+1})\widehat{\theta}_t \\
\implies &\mathbb{E}[\widehat{\theta}_{t+1}] = \alpha_{t+1}\theta^* + (1-\alpha_{t+1})\mathbb{E}[\widehat{\theta}_t] \\
\implies &\mathbb{E}[\widehat{\theta}_t] = \theta^*, \quad \text{using } \mathbb{E}[\hat{\theta}_0] = \theta^*.
\end{aligned}$$

Also, we have $\mathsf{Var}(\mathbb{E}[\widehat{\theta}_{t+1}(k) \mid \widehat{\theta}_t(k)]) = (1-\alpha_{t+1})^2\mathsf{Var}(\widehat{\theta}_t(k))$. Additionally,

$$\mathsf{Var}(\widehat{\theta}_{t+1}(k)|\widehat{\theta}_t(k)) = \frac{1}{n_{t+1}}(\alpha_{t+1}\theta^*(k) + (1-\alpha_{t+1})\widehat{\theta}_t(k))(1 - \alpha_{t+1}\theta^*(k) - (1-\alpha_{t+1})\widehat{\theta}_t(k))$$

Simplifying and taking expectation, we get

$$\mathbb{E}[\mathsf{Var}(\widehat{\theta}_{t+1}(k)|\widehat{\theta}_t(k))] = \frac{1}{n_{t+1}}\left[\theta^*(k)(1-\theta^*(k)) - (1-\alpha_{t+1})^2\mathsf{Var}(\widehat{\theta}_t(k))\right].$$

Now using the total law of variance, we note that the variance of estimate $\widehat{\theta}_{t+1}(k)$ satisfies the following recursion,

$$\begin{aligned}
&\mathsf{Var}(\widehat{\theta}_{t+1}(k)) = \mathbb{E}[\mathsf{Var}(\widehat{\theta}_{t+1}(k) \mid \widehat{\theta}_t)] + \mathsf{Var}(\mathbb{E}[\widehat{\theta}_{t+1}(k) \mid \widehat{\theta}_t]) \\
&= \frac{1}{n_{t+1}} \left(\theta^*(k)(1-\theta^*(k)) - (1-\alpha_{t+1})^2 \mathsf{Var}\widehat{\theta}_t(k) \right) + (1-\alpha_{t+1})^2 \mathsf{Var}\widehat{\theta}_t(k) \\
&= \frac{\theta^*(k)(1-\theta^*(k))}{n_{t+1}} + \frac{n_{t+1}-1}{n_{t+1}} (1-\alpha_{t+1})^2 \mathsf{Var}\widehat{\theta}_t(k).
\end{aligned}$$

Moreover,

$$\begin{aligned}
R_t &= \mathbb{E}\|\hat{\theta}_t - \theta^*\|_2^2 \\
&= \sum_{k=1}^{K} \mathbb{E}\left[\left(\hat{\theta}_t(k) - \theta^*(k) \right)^2 \right] \\
&= \sum_{k=1}^{K} \mathsf{Var}\widehat{\theta}_t(k).
\end{aligned}$$

Hence summing the recursion for $\mathsf{Var}\widehat{\theta}_{t+1}(k)$ over $k \in [K]$, we get the desired recursion for R_t. $\square$

Proof of Corollary 3.2. For Case 1 with fixed number of samples $n_t = n$ (and $n_0 = n$, so that $n_0 R_0 / n_t = R_0$) and $\alpha_t = \alpha$, the recursion from Theorem 3.1 simply becomes

$$R_t = a + bR_{t-1},$$

with

$$a = R_0 \text{ and } b = \frac{n-1}{n}(1-\alpha)^2 < 1.$$

Now solving the linear recursion

$$\begin{aligned}
&R_t - l = b(R_{t-1} - l) \\
&l - b\lambda = a \implies l = \frac{a}{1-b} \\
&R_t = l + b^t (R_0 - l)
\end{aligned}$$

$$\begin{aligned}
&\implies R_t \to l \text{ as } t \to \infty \\
&\text{with } l = R_0 \left(1 - \frac{n-1}{n}(1-\alpha)^2 \right)^{-1}.
\end{aligned}$$

Now notice if $\alpha_t > \alpha > 0$ for all $t \geq 1$, R_t would decay faster than what it would with fixed $\alpha_t = \alpha$. Hence the above limit for R_t with constant $\alpha_t = \alpha$ would be a limiting upper-bound hold for any sequence of $(\alpha_t)_{t \geq 1}$ if $\alpha_t > \alpha$.

For Case 2 with varying sample size $(n_t)_{t \geq 1}$, note that

$$R_t \leq \frac{c_0}{n_t} + \left(1 - \frac{1}{n_t} \right) \epsilon R_{t-1}$$

where $c_0 = n_0 R_0$ and $\epsilon = (1-\alpha)^2 < 1$. The above gives

$$R_t < \frac{c_0}{n_t} + \epsilon R_{t-1}$$

which recursively gives

$$\begin{aligned}
R_t &< \frac{c_0}{n_t} + \epsilon \frac{c_0}{n_{t-1}} + \epsilon^2 \frac{c_0}{n_{t-2}} + \cdots + \epsilon^{t-1} \frac{c_0}{n_1} + \epsilon^t R_0 \\
&= c_0 \sum_{s=1}^{t} \frac{\epsilon^{t-s}}{n_s} + \epsilon^t R_0.
\end{aligned}$$

Thus, if $n_t \uparrow \infty$, then $R_t \to 0$ as $t \to \infty$.

Details: Enough to show $S_t = \sum_{s=1}^{t} \frac{\epsilon^{t-s}}{n_s} = \sum_{u=0}^{t-1} \frac{\epsilon^u}{n_{t-u}} \to 0$. Fix δ, and we show that there exists T such that for $t \geq T$, $S_t \leq \delta$. For fixed and large M, let $S_t = A_t + B_t$, where A_t is the sum from 0 to M, and B_t is the sum from $M+1$ to t. Now,

$$B_t = \sum_{u=M+1}^{t} \frac{\epsilon^u}{n_{t-u}} \leq \sum_{u=M+1}^{\infty} \epsilon^u = \frac{\epsilon^M}{1-\epsilon}$$

Choose M so that $\epsilon^M/(1-\epsilon) < \delta/2$ (recall $\epsilon < 1$). Now,

$$A_t = \sum_{u=0}^{M} \frac{\epsilon^u}{n_{t-u}} \leq \sum_{u=0}^{M} \frac{\epsilon^u}{m_t} \leq \frac{1}{m_t} . \frac{1}{1-\epsilon},$$

where $m_t = \min_{0 \leq u \leq M} n_{t-u}$. Since $n_t \to \infty$, $m_t \to \infty$, hence for sufficiently large $t \geq T$, $A_t \leq \delta/2$. Making $S_t \leq \delta$.

□

Proof of Corollary 3.6. If we have $\alpha_t = 0$ the recursion from Theorem 3.1 simplifies and then further telescoping gives

$$\begin{aligned} & n_t(R_t - R_{t-1}) = n_0 R_0 - R_{t-1} \\ \implies & n_0 R_0 \sum_{s \leq t} (1/2^{t-s} n_s) < R_t < n_0 R_0 \sum_{s \leq t} (1/n_s). \end{aligned}$$

□

Proof of Proposition 3.4. Part 1: Define $\xi_t > 0$ such that

$$\xi_t = 1 - (1-\alpha_t)^2, \ \forall t \geq 1$$

Note that $\xi_t = 2\alpha_t - \alpha_t^2$ decays at the same rate as α_t. Let's begin with the general recurrence relation of R_t. To simplify notations replace $(1-\alpha_t)^2$ by $1-\xi_t$ and then divide both sides by R_{t-1}.

$$\begin{aligned} R_t &= \frac{n_0 R_0}{n_t} + \frac{n_t - 1}{n_t}(1-\alpha_t)^2 R_{t-1} \\ \implies R_t &= \frac{n_0 R_0}{n_t} + \frac{(n_t-1)(1-\xi_t)}{n_t} R_{t-1} \\ \implies \frac{R_t}{R_{t-1}} &= \frac{n_0 R_0}{n_t} \frac{1}{R_{t-1}} + \frac{(n_t-1)(1-\xi_t)}{n_t} \\ \implies \frac{R_t}{R_{t-1}} &\leq \frac{n_0 R_0}{n_t} \frac{n_{t-1}}{n_0 R_0} + \frac{(n_t-1)(1-\xi_t)}{n_t} \\ &= \frac{n_{t-1} + n_t - \xi_t(n_t-1) - 1}{n_t} =: \eta_t. \end{aligned}$$

The inequality used above is because $\forall t \geq 0$,

$$R_t = \frac{n_0 R_0}{n_t} + \frac{n_t - 1}{n_t}(1-\alpha_t)^2 R_{t-1} \implies R_t \geq \frac{n_0 R_0}{n_t}.$$

Observe $\eta_t > 0$ and

$$\begin{aligned} \eta_t < 1 &\iff n_{t-1} + n_t - \xi_t(n_t-1) - 1 < n_t \\ &\iff n_t > \frac{n_{t-1} - 1}{\xi_t} + 1, \end{aligned}$$

given ξ_t we recursively construct n_t to satisfy the above inequality. Now, telescoping products directly gives

$$\begin{aligned} & \frac{R_t}{R_{t-1}} \leq \frac{n_{t-1} + n_t - \xi_t(n_t-1) - 1}{n_t} =: \eta_t \\ \implies & R_t \leq \eta_t R_{t-1} \implies R_t < R_{t-1}. \end{aligned}$$

Thus growing the number of samples n_t satisfies the above recursive bound for all $t \geq 1$, then $(R_t)_{t\geq 0}$ is a decreasing sequence. Now as R_t is also lower-bounded by 0, we can conclude

$$\lim_{t\to\infty} R_t < R_0,$$

due to the monotone convergence theorem.

Part 2: Let us denote

$$a_t := \frac{n_0}{n_t} \to 0$$
$$b_t := \frac{n_t - 1}{n_t}(1-\alpha_t)^2 \in (0,1).$$

The recurrence takes the standard non-homogeneous linear form

$$R_t = a_t R_0 + b_t R_{t-1},$$

which can be unfolded to give

$$R_t = R_0 \underbrace{\sum_{k=0}^{t-1} a_{t-k} \prod_{j=t-k+1}^{t} b_j}_{s_t} + R_0 \underbrace{\prod_{j=1}^{t} b_j}_{p_t}.$$

We focus on the two terms individually and show that they go to 0.

Let us focus on the second product term p_t. Note that

$$p_t = \prod_{j=1}^{t} b_j = \left(\prod_{j=1}^{t} \frac{n_j - 1}{n_j}\right)\left(\prod_{j=1}^{t}(1-\alpha_j)^2\right)$$
$$= \prod_{j=1}^{t}\left(1 - \frac{1}{n_j}\right) \times \prod_{j=1}^{t}(1-\alpha_j)^2.$$

Both the products are bounded by 1 trivially. The first product goes to 0 if $\sum_t 1/n_t = \infty$, since $\log(1 - 1/n_t) \leq -1/n_t$. The second product goes to 0 if $\sum_t \alpha_t = \infty$, since $\log(1-\alpha_j)^2 \leq -2\alpha_j$ for $\alpha_j \in (0,1)$. Thus the term p_t goes to 0 if

$$\sum_t 1/n_t = \infty \quad \text{OR} \quad \sum_t \alpha_t = \infty.$$

Now, consider the first term, which is the sum s_t. Substituting $m = t - k$, we have

$$s_t = \sum_{m=1}^{t} a_m \prod_{j=m+1}^{t} b_j$$
$$= \sum_{m=1}^{t} \frac{n_0}{n_m} \prod_{j=m+1}^{t}\left(1 - \frac{1}{n_j}\right)(1-\alpha_j)^2.$$

Using $\log(1-x) \leq -x$ for $x \in (0,1)$, we obtain

$$\log(b_i) \leq -\frac{1}{n_i} - 2\alpha_i$$
$$\Rightarrow \prod_{j=m+1}^{t} b_j \leq \exp\left(-\sum_{j=m+1}^{t}\left(\frac{1}{n_j} + 2\alpha_j\right)\right)$$
$$\leq \exp\left(-2\sum_{j=m+1}^{t}\alpha_j\right).$$

Thus, (ignoring the n_0 fixed term)

$$s_t \le \sum_{m=1}^{t} \frac{1}{n_m} \exp\left(-2 \sum_{j=m+1}^{t} \alpha_j\right) =: \bar{s}_t.$$

Now, fix $\epsilon > 0$. We show that $\bar{s}_t < \epsilon$ for sufficiently large t. For this, we split the sum into two parts, for some large constant $J > 0$ (independent of t) to be specified later.

$$\bar{s}_t = \sum_{j=1}^{J} \frac{1}{n_j} \exp\left(-2 \sum_{i=j+1}^{t} \alpha_i\right) + \sum_{j=J+1}^{t} \frac{1}{n_j} \exp\left(-2 \sum_{i=j+1}^{t} \alpha_i\right) =: I_1 + I_2.$$

For I_1, this has fixed number of J terms. For each $j \in [1, J]$, the part in the exponential $\sum_{i=j+1}^{t} \alpha_i \to \infty$ as $t \to \infty$ with j fixed. Thus, there is large enough T such that for $t \ge T$, $I_1 \le \epsilon/2$.

Now, the constant J is chosen such that

$$\sum_{j=J+1}^{\infty} \frac{1}{n_j} < \epsilon/2,$$

which is possible since $\sum_t 1/n_t < \infty$. Since $\alpha_i > 0$, each exponential factor is at most 1, and hence for every $t > J$,

$$I_2 = \sum_{j=J+1}^{t} \frac{1}{n_j} \exp\left(-2 \sum_{i=j+1}^{t} \alpha_i\right) \le \sum_{j=J+1}^{\infty} \frac{1}{n_j} < \epsilon/2.$$

Note that J depends only on ϵ and (n_t), so the choice of T for I_1 above is valid. Thus, gives $\bar{s}_t \le \epsilon$. Since ϵ was arbitrary, we conclude $s_t \le \bar{s}_t \to 0$ as $t \to \infty$. Thus, $R_t = R_0(p_t + s_t) \to 0$. □

B.1 Accumulation and mixture

Consider the setting in Section 3.2. Recall, at iteration $t = 0$, we obtain $\widehat{\theta}_0$ based on n_0 i.i.d. samples from the true data distribution P_{θ^*}. Subsequently, at iteration $t \ge 1$, we train the model to obtain $\widehat{\theta}_t$ based on $n_t = n_{t-1} + m_t$ samples, containing the n_{t-1} samples that were used to train the model at iteration $t - 1$ (accumulation part), and m_t fresh (unseen) samples generated conditionally i.i.d. (given $\hat{\theta}_{t-1}$) from

$$Q_t = \alpha_t P_{\theta^*} + (1 - \alpha_t)\widehat{P}_{t-1}.$$

Denoting $R_t = \mathbb{E}\|\hat{\theta}_t - \theta^*\|_2^2$ as before, the following result captures the recursion followed by R_t. We assume that we use MLE on the entire training dataset to obtain $\hat{\theta}_t$, which implies that the model training is agnostic to the nature of the source of the particular samples.

Theorem B.1. *In the above setting, the sequence $(R_t)_{t\ge0}$ satisfies the following recurrence*

$$R_t = \left(\gamma_t^2 - \frac{m_t(1-\alpha_t)^2}{n_t^2}\right) R_{t-1} + \frac{m_t n_0}{n_t^2} R_0,$$

where the contraction factor γ_t is given by

$$\gamma_t = 1 - \frac{\alpha_t m_t}{n_t} \in (0, 1].$$

Proof of Theorem B.1. Let $\mathcal{F}_t$ be the natural filtration generated by all accumulated samples up to time t. Thus, $\hat{\theta}_t$ is $\mathcal{F}_t$-measurable. Let $\bar{Z}_t$ be the empirical mean vector of the *new* batch (of size m_t) during the training in iteration t. Then, we have

$$\hat{\theta}_t = \frac{n_{t-1}}{n_t}\hat{\theta}_{t-1} + \frac{m_t}{n_t}\bar{Z}_t.$$

Note that $\mathbb{E}[\bar{Z}_t \mid \mathcal{F}_{t-1}] = Q_t$. Hence, subtracting θ^* from both sides and taking the conditional expectation, we get

$$\mathbb{E}[\hat{\theta}_t - \theta^* \mid \mathcal{F}_{t-1}] = \frac{n_{t-1}}{n_t}(\hat{\theta}_{t-1} - \theta^*) + \frac{m_t}{n_t}(Q_t - \theta^*).$$

Now,

$$Q_t - \theta^* = \alpha_t\theta^* + (1-\alpha_t)\hat{\theta}_{t-1} - \theta^* = (1-\alpha_t)(\hat{\theta}_{t-1} - \theta^*).$$

Plugging this back and using $n_t = n_{t-1} + m_t$, we get

$$\mathbb{E}[\hat{\theta}_t - \theta^* \mid \mathcal{F}_{t-1}] = \left(\frac{n_{t-1} + m_t(1-\alpha_t)}{n_t}\right)(\hat{\theta}_{t-1} - \theta^*) = \underbrace{\left(1 - \frac{\alpha_t m_t}{n_t}\right)}_{\gamma_t}(\hat{\theta}_{t-1} - \theta^*).$$

Rewriting the error vector as a zero-mean fluctuation around its conditional mean, we obtain

$$\hat{\theta}_t - \theta^* = \gamma_t(\hat{\theta}_{t-1} - \theta^*) + \frac{m_t}{n_t}(\bar{Z}_t - Q_t).$$

Taking the squared ℓ_2-norm and expanding

$$\|\hat{\theta}_t - \theta^*\|_2^2 = \gamma_t^2\|\hat{\theta}_{t-1} - \theta^*\|_2^2 + \frac{m_t^2}{n_t^2}\|\bar{Z}_t - Q_t\|_2^2 + \frac{2\gamma_t m_t}{n_t}\langle\hat{\theta}_{t-1} - \theta^*, \bar{Z}_t - Q_t\rangle.$$

Take conditional expectation $\mathbb{E}[\cdot \mid \mathcal{F}_{t-1}]$ on both sides. Since $\hat{\theta}_{t-1} - \theta^*$ is $\mathcal{F}_{t-1}-$measurable and $\mathbb{E}[\bar{Z}_t - Q_t \mid \mathcal{F}_{t-1}] = 0$, the cross term vanishes. This yields

$$\mathbb{E}[\|\hat{\theta}_t - \theta^*\|_2^2 \mid \mathcal{F}_{t-1}] = \gamma_t^2\|\hat{\theta}_{t-1} - \theta^*\|^2 + \frac{m_t^2}{n_t^2}\mathbb{E}[\|\bar{Z}_t - Q_t\|_2^2 \mid \mathcal{F}_{t-1}].$$

Now, $\bar{Z}_t = \frac{1}{m_t}\sum_{i=1}^{m_t} Z_{t,i}$, where $Z_{t,i}$ are categorical random variables and i.i.d. (conditionally) from Q_t. Thus,

$$\mathbb{E}[\|\bar{Z}_t - Q_t\|_2^2 \mid \mathcal{F}_{t-1}] = \frac{1}{m_t}\mathsf{tr}\left(\mathsf{Var}(Z_{t,1} \mid \mathcal{F}_{t-1})\right) = \frac{1 - \|Q_t\|_2^2}{m_t}.$$

Substituting and taking another expectation, we can use the law of total expectation to obtain the recursion in terms of $V_t = \mathbb{E}[1 - \|Q_t\|_2^2]$:

$$R_t = \gamma_t^2 R_{t-1} + \frac{m_t}{n_t^2}V_t.$$

Now, $V_t = \mathbb{E}[1 - \|Q_t\|_2^2]$. Expanding $Q_t = \theta^* + (1-\alpha_t)(\hat{\theta}_{t-1} - \theta^*)$ we get

$$\|Q_t\|_2^2 = \|\theta^*\|_2^2 + 2(1-\alpha_t)\langle\theta^*, \hat{\theta}_{t-1} - \theta^*\rangle + (1-\alpha_t)^2\|\hat{\theta}_{t-1} - \theta^*\|_2^2.$$

Since $\mathbb{E}[\hat{\theta}_{t-1}] = \theta^*$, expectation of the inner product term is 0. Hence, we have

$$\mathbb{E}[\|Q_t\|_2^2] = \|\theta^*\|_2^2 + (1-\alpha_t)^2 R_{t-1}.$$

Recalling $1 - \|\theta^*\|_2^2 = n_0 R_0$, we get the recursion for V_t as

$$V_t = n_0 R_0 - (1-\alpha_t)^2 R_{t-1}.$$

Now, plugging this back, we have

$$\begin{aligned}
R_t &= \gamma_t^2 R_{t-1} + \frac{m_t}{n_t^2}V_t \\
&= \gamma_t^2 R_{t-1} + \frac{m_t}{n_t^2}\left(n_0 R_0 - (1-\alpha_t)^2 R_{t-1}\right) \\
&= \left(\gamma_t^2 - \frac{m_t(1-\alpha_t)^2}{n_t^2}\right)R_{t-1} + \frac{m_t n_0}{n_t^2}R_0.
\end{aligned}$$

Note: If $n_t = m_t$, then $\gamma_t = 1-\alpha_t$. Then the coefficient of R_{t-1} becomes $(1-\alpha_t)^2 - (1-\alpha_t)^2/n_t = (1 - 1/n_t)(1-\alpha_t)^2$. And the coefficient of R_0 boils down to n_0/n_t, thus matching the recursion in Theorem 3.1. $\square$

Limiting structure of R_t under various regimes for the setting in Section 3.2

We discuss these in various steps. We consider the following cases: (i) Pure synthetic $\alpha_t = 0$, (ii) Linear growth rate $m_t = m$ fixed, and (iii) Geometric growth rate $m_t = cn_{t-1}$ for fixed $c > 0$. As we discuss, it is easy to characterize cases where consistency occurs. However, for cases where $R_\infty \neq 0$, checking whether we are in a total collapse setting or iterative improvement depends on the nuanced balance of n_t and α_t. To begin, recall the recursion

$$R_t = A_t R_{t-1} + B_t R_0$$

where

$$A_t = \gamma_t^2 - \frac{m_t(1-\alpha_t)^2}{n_t^2} = \frac{n_{t-1}^2 + 2n_{t-1}m_t(1-\alpha_t) + m_t(m_t-1)(1-\alpha_t)^2}{n_t^2},$$
$$B_t = \frac{m_t n_0}{n_t^2}$$

with $\gamma_t = 1 - \alpha_t m_t/n_t$. By unrolling this linear recurrence, we obtain the explicit closed form solution

$$R_t = R_0 \left(\prod_{j=1}^{t} A_j + \sum_{j=1}^{t} B_j \prod_{k=j+1}^{t} A_k \right)$$

with the convention $\prod_{k=t+1}^{t} A_k = 1$.

(i) Pure synthetic $\alpha_t = 0$: Here we have the simplified forms $A_t = 1 - m_t/n_t^2$ and B_t as before, leading to

$$R_t = (1 - m_t/n_t^2)R_{t-1} + m_t n_0 R_0/n_t^2.$$

Subtract $n_0 R_0$ from both sides and unroll the recursion to get

$$R_t - n_0 R_0 = (R_0 - n_0 R_0) \prod_{j=1}^{t} (1 - m_j/n_j^2) = -(n_0 - 1)R_0 \prod_{j=1}^{t} (1 - m_j/n_j^2).$$

Since $n_j > m_j \geq 1$, we have $n_j^2 > m_j$, meaning $0 < 1 - m_j/n_j^2 < 1$. Thus, for any $n_0 > 1$,

$$-(n_0 - 1)R_0 < 0 \Rightarrow R_t - n_0 R_0 < 0 \Rightarrow R_t < n_0 R_0 \quad \text{for all } t \geq 0.$$

Now, from the original recursion, we can write

$$R_t - R_{t-1} = (1 - m_t/n_t^2)R_{t-1} + m_t n_0 R_0/n_t^2 - R_{t-1} = \frac{m_t}{n_t^2}(n_0 R_0 - R_{t-1}).$$

Thus for any $n_0 > 1$, we have $R_t > R_{t-1} \geq 0$ for all t, since $n_0 R_0 > R_{t-1}$. Thus, the risk is monotonically increasing.

Special subcase: $m_t = m$: In this case, we are working with accumulation from the previous time point, but the new data added in iteration t is of fixed size m and is purely synthetic. In this case, we can show a much tighter upper bound compared to the $R_t < n_0 R_0$ that we have already established. We can use Weierstrass product inequality $\prod_{j=1}^{t}(1 - x_j) \geq 1 - \sum_{j=1}^{t} x_j$ to obtain:

$$R_t = n_0 R_0 - (n_0 - 1)R_0 \prod_{j=1}^{t} \left(1 - \frac{m}{n_j^2}\right) \leq R_0 + (n_0 - 1)R_0 \sum_{j=1}^{t} \frac{m}{(n_0 + mj)^2}.$$

Since $f(x) = m/(n_0 + mx)^2$ is strictly decreasing for $x \geq 0$, we can bound

$$\sum_{j=1}^{t} \frac{m}{n_j^2} \leq \int_0^t \frac{m}{(n_0 + mx)^2} dx = \frac{1}{n_0} - \frac{1}{n_t}.$$

Thus, we obtain

$$R_t \leq R_0 + (n_0 - 1)R_0(1/n_0 - 1/n_t) = \left(2 - \frac{1}{n_0} - \frac{n_0 - 1}{n_t}\right) R_0.$$

Taking $t \to \infty$, we have the uniform asymptotic cap:

$$R_\infty \leq \left(2 - \frac{1}{n_0}\right) R_0 < 2R_0.$$

In the special subcase $n_0 = m$, $n_j = n_0(j+1)$, and we can instead use $\sum_{j=1}^{\infty} 1/(1+j)^2 = \pi^2/6 - 1$ instead of the integral bound, to obtain

$$R_t \leq R_0 \left(1 + \frac{n_0 - 1}{n_0}\left(\frac{\pi^2}{6} - 1\right)\right) < 1.645 R_0.$$

(ii) Linear Data Growth: For $m_t = m$, we have $n_t = n_0 + mt = mt(1 + \mathcal{O}(1/t))$. Expanding A_t and B_t asymptotically

$$A_t = \left(1 - \frac{\alpha_t m}{n_t}\right)^2 - \frac{m(1-\alpha_t)^2}{n_t^2} = 1 - \frac{2\alpha_t}{t} + \mathcal{O}(1/t^2)$$
$$B_t = \frac{m n_0}{n_t^2} = \frac{n_0/m}{t^2} + \mathcal{O}(1/t^3).$$

Now, we divide into two cases: (a) Constant $\alpha_t = \alpha > 0$ and (b) Decaying $\alpha_t \sim t^{-\beta}$ for $\beta > 0$.

For the former case $\alpha_t = \alpha$, the product term $P_{j,t} = \prod_{k=j+1}^{t} A_k$ satisfies

$$P_{j,t} = \exp\left(-\sum_{k=j+1}^{t}\left(\frac{2\alpha}{k} + \mathcal{O}(1/k^2)\right)\right) = \left(\frac{j}{t}\right)^{2\alpha}(1 + \mathcal{O}(1/j)).$$

Plugging this and $B_j \sim n_0/mj^2$ into the explicit solution, we have

$$R_t = R_0 P_{0,t} + R_0 \sum_{j=1}^{t} B_j P_{j,t} = \mathcal{O}(t^{-2\alpha}) + \frac{n_0 R_0}{m t^{2\alpha}} \sum_{j=1}^{t} j^{2\alpha-2}(1 + \mathcal{O}(1/j)).$$

Evaluating the integral approximation $\sum_{j=1}^{t} j^{2\alpha-2}$, we have

1. Case $\alpha > 1/2$: Here $2\alpha - 2 > -1$, and $\sum_{j=1}^{t} j^{2\alpha-2} = \frac{t^{2\alpha-1}}{2\alpha-1} + \mathcal{O}(1)$. Multiplying by $t^{-2\alpha}$, we get $R_t = \mathcal{O}(1/t)$.
2. Case $\alpha = 1/2$: Hence $2\alpha - 2 = -1$ and $\sum_{j=1}^{t} j^{-1} = \log t + \mathcal{O}(1)$. Thus, $R_t = \mathcal{O}(\log t/t)$.
3. Case $\alpha < 1/2$: Here $2\alpha - 2 < -1$ and $\sum_{j=1}^{t} j^{2\alpha-2} = \mathcal{O}(1)$, which gives $R_t = \mathcal{O}(t^{-2\alpha})$

Thus, in all three cases, if $\alpha_t = \alpha > 0$ constant, then $R_t \to 0$ as $t \to \infty$.

Now, consider the decaying case $\alpha_t \sim t^{-\beta}$. Here

$$A_t = 1 - \frac{2\alpha_t}{t} - \frac{1}{mt^2} + \mathcal{O}(\alpha_t/t^2) = 1 - \frac{2c}{t^{1+\beta}} - \frac{1}{mt^2} + \mathcal{O}(1/t^{2+\beta}).$$

Since both the exponents $1 + \beta > 1$ and $2 > 1$, we have $\sum_{t=1}^{\infty}(1 - A_t) < \infty$, the infinite product $\prod_{j=1}^{\infty} A_j$ converges to a strictly positive constant $C_A > 0$. Thus, $R_t \geq R_0 \prod_{j=1}^{t} A_j \to R_0 C_A > 0$. So, $R_\infty \neq 0$. We discuss this case in slightly more detail to understand when $R_\infty < R_0$ (iterative improvement) and when $R_\infty > R_0$ (collapse).

Recall the asymptotic limit in this case

$$R_\infty = R_0\left(P_0 + \sum_{j=1}^{\infty} B_j P_j\right), \quad \text{where } P_j = \prod_{k=j+1}^{\infty} A_k.$$

Since $\prod_{j=1}^{\infty} A_j$ is convergent, its tail satisfies $P_j \to 1$ as $j \to \infty$. Using the telescoping property $P_j - P_{j-1} = P_j(1 - A_j)$, we have

$$\sum_{j=1}^{\infty} P_j(1 - A_j) = \lim_{M\to\infty} P_M - P_0 = 1 - P_0 \Rightarrow P_0 = 1 - \sum_{j=1}^{\infty} P_j(1 - A_j).$$

Substituting back, we get

$$R_\infty - R_0 = R_0 \sum_{j=1}^{\infty} P_j \underbrace{(B_j - (1 - A_j))}_{=:D_j}.$$

Substituting the definitions of A_j and B_j with $m_j = m$, we get the step different weight

$$D_j = \frac{m}{n_j^2} \left[(n_0 - 1) - 2\alpha_j (n_j - 1) + (m - 1)\alpha_j^2 \right].$$

Since $P_j > 0$ for all j, the sign of $R_\infty - R_0$ depends on the bracketed term (to first order):

$$g(j) = \underbrace{(n_0 - 1)}_{\text{accumulation drift}} - \underbrace{2\alpha_j (n_j - 1)}_{\text{synthetic filtering}} + \mathcal{O}(\alpha_j^2).$$

For $n_j \approx mj$ and $\alpha_t \sim ct^{-\beta}$, the filtering term scales as $2\alpha_j(n_j - 1) \approx cmj^{1-\beta}$. This brings us to different cases depending on the rate of decay β.

If $\beta > 1$ (fast decay), then $R_\infty > R_0$ as $j^{1-\beta} \to 0$. Synthetic data vanished too fast to stop accumulation drift.

If $\beta < 1$ (slow decay), then $R_\infty < R_0$ as $j^{1-\beta} \to \infty$. Here we see iterative improvement, however $R_\infty > 0$.

Finally, in the critical decay case $\beta = 1$, i.e., $\alpha_t \sim c/t$, we have $2\alpha_j(n_j - 1) \to 2cm$. Thus asymptotically, $g(\infty) = (n_0 - 1) - 2cm$. Hence, the result depends on whether $c < (n_0 - 1)/2m$. If so, then $R_\infty > R_0$, while if $c > (n_0 - 1)/2m$, then $R_\infty < R_0$.

(iii) Geometric Data Growth: For $m_t = cn_{t-1}$ for a fixed $c > 0$, we have $n_t = n_0(1 + c)^t$ and $m_t/n_t = c/(1 + c)$ fixed. Evaluating A_t and B_t in this case gives

$$\begin{aligned}
\gamma_t &= 1 - \alpha_t \frac{c}{c+1} = \frac{1 + c(1 - \alpha_t)}{1 + c} \\
A_t &= \gamma_t^2 - \frac{c(1 - \alpha_t)^2}{(1 + c)^{t+1} n_0} \\
B_t &= \frac{c}{(1 + c)^{t+1}}.
\end{aligned}$$

Again, we divide into two cases (a) constant $\alpha_t = \alpha > 0$ and (b) decaying $\alpha_t \sim t^{-\beta}$.

In the constant case $\alpha_t = \alpha$, since $\alpha > 0$, we have $\gamma = (1 + c(1 - \alpha))/(1 + c) < 1$. Now, $A_t = \gamma^2(1 - \mathcal{O}((1+c)^{-t}))$. Recall $R_t = R_0(P_{0,t} + \sum_{j=1}^t B_j P_{j,t})$. The first part $P_0 = \prod_{j=1}^t A_j = \Theta(\gamma^{2t})$ since $\sum_{j=1}^\infty (1 + c)^{-j} < \infty$. Further, $P_{j,t} = \prod_{k=j+1}^t A_k = \Theta(\gamma^{2(t-j)})$.

Substituting $B_j = c/(1 + c)^{t+1}$ into the second part, we get

$$S_t = \sum_{j=1}^t B_j P_{j,t} = \sum_{j=1}^t \frac{c}{(1 + c)^{j+1}} \gamma^{2(t-j)} = \mathcal{O}\left(\gamma^{2t} \sum_{j=1}^t \left(\frac{1}{\gamma^2(1 + c)} \right)^j \right).$$

Let $x = 1/\gamma^2(1 + c)$. If $x < 1$, $\sum_{j=1}^t x^j = \mathcal{O}(1)$, leading to $S_t = \mathcal{O}(\gamma^{2t})$. If $x > 1$, then the sum grows as $\mathcal{O}(x^t) = \mathcal{O}((\gamma^2(1 + c))^{-t})$, and hence, $S_t = \mathcal{O}((1 + c)^{-t}) = \mathcal{O}(1/n_t)$. If $x = 1$, then $S_t = \mathcal{O}(\log n_t/n_t)$. Thus, in all cases $R_t \to 0$.

In the decaying $\alpha_t \sim t^{-\beta}$ case, we have $\gamma_t = 1 - \frac{c}{1+c}\alpha_t$, so $A_t = 1 - \frac{2c}{1+c}\alpha_t + \mathcal{O}(\alpha_t^2 + (1 + c)^{-t})$. The term $P_{0,t} = \prod_{k=1}^t A_k$ behaves as

$$P_{0,t} = \exp\left(-\frac{2c}{1 + c} \sum_{k=1}^t \alpha_k + \mathcal{O}(1) \right).$$

Thus, if $\beta \leq 1$, $\sum_k \alpha_k = \infty$, implying $P_{0,t} \to 0$. Since $B_j = c/(1+c)^t$ is exponentially summable ($\sum_{j=1}^{\infty} B_j = 1/(1+c) < \infty$, and $P_{j,t} \leq 1$, by dominated convergence theorem,

$$\lim_{t\to\infty} \sum_{j=1}^{t} B_j P_{j,t} = \sum_j B_j \left(\lim_{t\to\infty} P_{j,t}\right) = \sum_j B_j.0 = 0.$$

Thus, in this case, $R_t = R_0 P_{0,t} + R_0 \sum_{j=1}^{t} B_j P_{j,t} \to 0$ as $t \to \infty$.

If $\beta > 1$, then $\sum_k \alpha_k < \infty$, implying $P_{0,t} \to C_A > 0$. Hence $R_t \geq R_0 C_A > 0$. Again, we go back to the telesoping identity

$$R_\infty - R_0 = R_0 \sum_{j=1}^{\infty} P_j D_j, \quad \text{where } D_j = B_j - (1 - A_j).$$

In the current setting, plugging in the forms for A_j, γ_j and B_j, we get

$$D_j = \underbrace{\frac{c(n_0-1)}{n_0(1+c)^{j+1}}}_{\text{accumulation drift}} - \underbrace{\frac{2c\alpha_j}{1+c}}_{\text{synthetic filtering}} + \mathcal{O}\left(\frac{\alpha_j}{(1+c)^j} + \alpha_j^2\right).$$

Substituting $\alpha_j \sim c_0 j^{-\beta}$ and evaluating the parts of the sum $\sum_j P_j D_j$, we get the following. We also use $P_j = 1 + \mathcal{O}(j^{1-\beta})$ since $P_j = \prod_{k=j+1}^{\infty} A_k$ with $A_k = 1 - \frac{2cc_0}{1+c} k^{-\beta} + \mathcal{O}(k^{-2\beta} + (1+c)^{-k})$.

- Positive drift $S_+ = \sum_{j=1}^{\infty} P_j \frac{c(n_0-1)}{n_0(1+c)^{j+1}} \approx \frac{n_0-1}{n_0(1+c)}$ (geometric series sum).
- Negative filtering sum $S_- = \sum_{j=1}^{\infty} P_j \frac{2cc_0 j^{-\beta}}{1+c} \approx \frac{2cc_0}{1+c}\zeta(\beta)$, where $\zeta(\beta) = \sum_{j=1}^{\infty} j^{-\beta}$.

The condition for iterative improvement ($R_\infty < R_0$) boils down to $S_- > S_+$, which is satisfied when $c_0 > \frac{n_0-1}{2n_0 c\zeta(\beta)}$.

Note: In both the above, and in the linear data growth case (with $\beta = 1$), it is important to understand that the critical conditions derived ($c_0 > \frac{n_0-1}{2n_0 c\zeta(\beta)}$ in the former, and $c > (n_0-1)/2m$ in the latter) are only first-order (leading-asymptotic) thresholds, not exact critical thresholds. But they indeed show that in these cases, there is some critical threshold c^* such that $c_0 > c^*$ leads to iterative improvement (recall the c_0 controls the decay rate $\alpha_t \sim c_0 t^{-\beta}$) and vice-versa (and similarly for the linear data growth case).

The findings above are summarized in Proposition 3.7.

Remark B.2. To better appreciate all the different settings and cases, let us compare the various cases in the special case $\alpha_t = \alpha$ fixed. We only look at cases where the training data size n_t is either fixed or growing linearly.

1. No accumulation with $n_t = m$: Here $R_t \to R_0(1 - (1-1/m)(1-\alpha)^2)^{-1}$. Thus, with purely synthetic samples ($\alpha = 0$), we have $R_t \to mR_0$, while with fresh true samples ($\alpha_t = 1$) $R_t \to R_0$. In this case, of course the training data size is m at all iterations, hence the best possible behavior is achieving $R_0 = (1 - \|\theta^*\|_2^2)/m$.
2. Accumulation with $m_t = m$ and hence, $n_t = n_{t-1} + m = mt$: If only synthetic new samples are added in each iteration ($\alpha_t = 0$), then $R_0 < R_\infty < R_0(1 + (1-1/m)(\pi^2/6 - 1))$. If $\alpha > 0$, then $R_t \to 0$.
3. No accumulation with $n_t = mt$: $R_t \to 0$ as long as $\alpha > 0$. If $\alpha = 0$, then $R_t \to mR_0$. Note that though both this setting and the last use $n_t = mt$ total samples to train the model at iteration t, the current setting uses all mt previously unseen samples, while the previous case uses only m previously unseen samples, with the remaining $m(t-1)$ coming from the dataset used in the last iteration.

Thus, while accumulation helps to prevent the risk from going to the worst possible limit mR_0, by itself it cannot *improve* the model iteratively. On the other hand, having only a little proportion

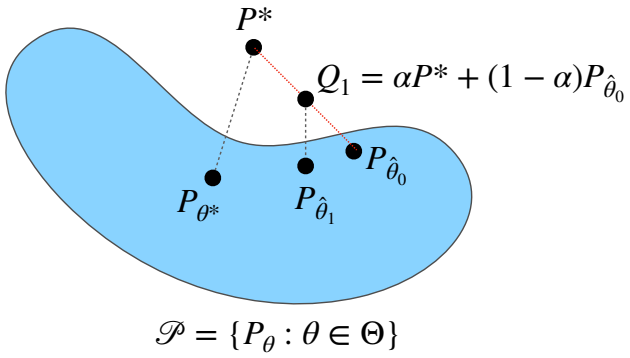


Figure 5: Visual Depiction of Theorem 4.2: Fresh information can improve estimation in the context of using MLE given a model class $\mathcal{P}$, where the MLE can be seen as estimating the KL-projection of the Q_1 on the model space. P_{θ^*} is the projection of the true data distribution P^*.

of fresh previously unseen samples from the true distribution iteratively *improves* the model, even achieving consistency without accumulation. When combined together, we see that consistency is achieved under extremely mild condition like only acquiring fixed size new dataset at each iteration (as in point 2 above). It is interesting to look at the cases $\alpha = 0$ from points 2 and 3 above: with accumulation we have $R_\infty < 2R_0$, while without accumulation, we have $R_\infty = mR_0$, which is much worse. However, although the total sample size is same in both cases, the setting with accumulation only has to deal with m previously unseen synthetic samples, while the case without accumulation is corrupted by all mt previously unseen synthetic samples, which explains the result.

C Proofs & Results in Section 4

C.1 An illustrative example

We first illustrate an example, which captures the main intuition of the two theorems in the section. Consider a simple setting with $\mathcal{P} = \{N(\mu, 1) : \mu \in \mathbb{R}\}$ with negative log-likelihood loss. From an empirical risk minimization perspective, this is equivalent to a loss $\ell(\theta, X) = (\theta - X)^2$ – however, it is better to think about it from a distribution estimation framework since typically the estimator must result in a generative model, to be used in the next stage. Given $X_1, \ldots, X_n$ at first stage with sample mean $\tilde{\theta}_0 = \bar{X}$, let $Y_1, \ldots, Y_m \sim P_{\tilde{\theta}_0}$. With accumulation, the dataset is D_1 consisting of all X and Y. The optimal estimator based on D_1, under the assumption that the data fully comes from a single unknown normal distribution, is $\hat{\theta}_1 := (n\bar{X} + m\bar{Y})/(n+m)$. Note that $D_1|D_0$ is free of P^*. Further,

$$\hat{\theta}_0 := \mathbb{E}[\hat{\theta}_1|D_0] = \frac{n}{n+m}\bar{X} + \frac{m}{n+m}\hat{\theta}_0 = \bar{X}$$

and the result says that $\hat{\theta}_0 = \bar{X}$ (which in this case is the same as $\tilde{\theta}_0$) is as good as $\hat{\theta}_1$. We further note that both estimators are unbiased, but variance of $\hat{\theta}_0$ is lower than that of $\hat{\theta}_1$ – thereby, $\hat{\theta}_0$ (based only on D_0) is a better estimator with respect to this loss. In other words, the additional next stage D_1 does not help in any way. Interestingly, the sample size m does not matter.

On the other hand, say we start with $\theta_0 \neq \theta^*$. Population distribution at next stage under a mixture setting is $Q_1 = \alpha N(\mu^*, 1) + (1-\alpha)N(\theta_0, 1)$ – note that this is not the same as $N(\alpha\mu^* + (1-\alpha)\theta_0, 1)$, in fact, $Q_1 \notin \mathcal{P}$. However, under the negative log-likelihood loss, the KL projection of Q_1 on the model space is $N(\alpha\mu^* + (1-\alpha)\theta_0, 1)$. Thus, $\theta_1 = \alpha\mu^* + (1-\alpha)\theta_0$. If $\alpha > 0$, θ_1 has a strictly better risk than θ_0 with respect to the true distribution $P^* = N(\mu^*, 1)$. At a sample level, say $\hat{\theta}_1^m$ is based on a finite sample of size m from Q_1. Then, we know that $\hat{\theta}_1^m \to \theta_1$ as $m \to \infty$, thereby, with sufficiently large m, the risk with $\hat{\theta}_1^m$ is better than θ_0. The precise size depends on α as well.

Lastly, we give an example, where without the condition $\mathcal{P}$ being convex, one might not improve for low α. Consider $\mathcal{P} = \{N(0, 1)\} \cup \{N(1, 1)\}$, and the true distribution be $N(0, 1)$. If $\theta_0 = 1$ (i.e.,

starting at $N(1,1)$), then at next stage $Q_1 = \alpha N(0,1) + (1-\alpha)N(1,1)$, whose projection is still $N(1,1)$ if

$$|(1-\alpha)-1| < |(1-\alpha)-0| \iff \alpha < 1/2.$$

Thus, in this case, with low α ($\alpha < 1/2$), the estimator is stuck at the sub-optimal θ_0.

Proof of Theorem 4.1. Let $\hat{\theta}_1$ be an estimator based on D_1. In that case, since $D_1|D_0$ is independent of P^*, $\mathbb{E}[\hat{\theta}_1|D_0]$ is a statistic (free of P^*), that is a measureable function of D_0 and since $D_1|D_0$ is known, this can be computed. Let $\hat{\theta}_0 = \mathbb{E}[\hat{\theta}_1|D_0]$. Then by Jensen's inequality, it follows that

$$\ell(\hat{\theta}_0, X) = \ell\left(\mathbb{E}[\hat{\theta}_1|D_0], X\right) \leq \mathbb{E}[\ell(\hat{\theta}_1, X)|D_0].$$

Taking expectation on both sides give the desired inequality. □

Proof of Theorem 4.2. Firstly, note that $\mathcal{R}(\theta, P_{\theta'}) = \mathbb{E}_{X\sim p_{\theta'}}[-\log p_\theta(X)] = \mathsf{KL}(P_{\theta'}\|P_\theta) + H(P_{\theta'})$, where H is the entropy. Thus, for fixed $\theta' = \theta_0$, the risk $\mathcal{R}(\theta, P_{\theta_0})$ is minimized at P_{θ_0}, i.e., $\mathcal{R}(\theta_0, P_{\theta_0}) \leq \mathcal{R}(\theta_1, P_{\theta_0})$. For the first part, if $\theta_1 = \theta_0$, there is nothing to show. Otherwise, note that by the definition of θ_1

$$\begin{aligned}
&\mathcal{R}(\theta_1, Q_1) \leq \mathcal{R}(\theta_0, Q_1) \\
\Rightarrow\ &\alpha\mathcal{R}(\theta_1, P^*) + (1-\alpha)\mathcal{R}(\theta_1, P_{\theta_0}) \\
&\leq \alpha\mathcal{R}(\theta_0, P^*) + (1-\alpha)\mathcal{R}(\theta_0, P_{\theta_0}).
\end{aligned}$$

Now, using the fact that $\mathcal{R}(\theta_0, P_{\theta_0}) \leq \mathcal{R}(\theta_1, P_{\theta_0})$, the above inequality shows that if $\alpha > 0$, then $\mathcal{R}(\theta_1, P^*) \leq \mathcal{R}(\theta_0, P^*)$. □

Strict improvement. We now give conditions under which the inequality in Theorem 4.2 is strict. Below we identify parameters with the distributions they index, and call $\mathcal{P}$ convex if $(1-s)P + sQ \in \mathcal{P}$ for all $P, Q \in \mathcal{P}$ and $s \in [0,1]$.

Proposition C.1. *In the setting of Theorem 4.2, let $\alpha \in (0,1]$ and $\theta^* \in \arg\min_\theta \mathcal{R}(\theta, P^*)$. Suppose that $\mathcal{P}$ is convex, that $\mathcal{R}(\theta_0, P^*) > \mathcal{R}(\theta^*, P^*)$, that $P_{\theta^*} \ll P_{\theta_0}$, and that $\mathcal{R}(\theta_0, P^*)$ and $\mathsf{KL}(P_{\theta_0}\|P_{\theta^*})$ are finite. Then $\mathcal{R}(\theta_1, P^*) < \mathcal{R}(\theta_0, P^*)$.*

Proof. If $\alpha = 1$, then $Q_1 = P^*$ and $\mathcal{R}(\theta_1, P^*) = \min_\theta \mathcal{R}(\theta, P^*) < \mathcal{R}(\theta_0, P^*)$. Let $\alpha \in (0,1)$. By the proof of Theorem 4.2 and $\mathcal{R}(\theta, P_{\theta_0}) = \mathsf{KL}(P_{\theta_0}\|P_\theta) + H(P_{\theta_0})$,

$$\alpha\big[\mathcal{R}(\theta_1, P^*) - \mathcal{R}(\theta_0, P^*)\big] \leq -(1-\alpha)\big[\mathcal{R}(\theta_1, P_{\theta_0}) - \mathcal{R}(\theta_0, P_{\theta_0})\big] = -(1-\alpha)\mathsf{KL}(P_{\theta_0}\|P_{\theta_1}).$$

It therefore suffices to show $P_{\theta_1} \neq P_{\theta_0}$, i.e., that P_{θ_0} does not minimize $f(Q) := \alpha\mathcal{R}(Q, P^*) + (1-\alpha)\mathcal{R}(Q, P_{\theta_0})$ over $\mathcal{P}$. For $s \in [0,1]$ let $Q_s = (1-s)P_{\theta_0} + sP_{\theta^*} \in \mathcal{P}$, with density $q_s = (1-s)p_{\theta_0} + sp_{\theta^*}$. For any P, $s \mapsto \mathcal{R}(Q_s, P) = \mathbb{E}_P[-\log q_s(X)]$ is convex, since $-\log$ is convex.

Let $g(s) = \mathcal{R}(Q_s, P^*)$. By convexity, for $s \in (0,1]$, $\{g(s)-g(0)\}/s \leq g(1)-g(0) = \mathcal{R}(\theta^*, P^*) - \mathcal{R}(\theta_0, P^*) =: -\delta < 0$.

Let $h(s) = \mathcal{R}(Q_s, P_{\theta_0})$ and $r = p_{\theta^*}/p_{\theta_0}$ on $\{p_{\theta_0} > 0\}$. Then $\{h(s) - h(0)\}/s = \mathbb{E}_{P_{\theta_0}}[-\log(1 + s(r-1))/s]$. For each fixed r, $s \mapsto -\log(1+s(r-1))$ is convex and vanishes at $s = 0$, so the integrand is nondecreasing in s, decreases to $1-r$ as $s \downarrow 0$, and is bounded above by its value $-\log r$ at $s = 1$, which is integrable since $\mathbb{E}_{P_{\theta_0}}[-\log r] = \mathsf{KL}(P_{\theta_0}\|P_{\theta^*}) < \infty$. By monotone convergence, $\{h(s) - h(0)\}/s \to \mathbb{E}_{P_{\theta_0}}[1-r] = 1 - P_{\theta^*}(p_{\theta_0} > 0) = 0$, using $P_{\theta^*} \ll P_{\theta_0}$.

Combining, $\{f(Q_s) - f(Q_0)\}/s \leq -\alpha\delta + (1-\alpha)\{h(s) - h(0)\}/s < 0$ for all sufficiently small $s > 0$. Hence $f(Q_s) < f(P_{\theta_0})$, so P_{θ_0} does not minimize f, $P_{\theta_1} \neq P_{\theta_0}$ and $\mathsf{KL}(P_{\theta_0}\|P_{\theta_1}) > 0$, which gives $\mathcal{R}(\theta_1, P^*) < \mathcal{R}(\theta_0, P^*)$. □

The condition $P_{\theta^*} \ll P_{\theta_0}$ is used to control the first-order change of $\mathcal{R}(\cdot, P_{\theta_0})$: if P_{θ^*} puts mass where $p_{\theta_0} = 0$, moving from P_{θ_0} towards P_{θ^*} increases $\mathcal{R}(\cdot, P_{\theta_0})$ at first order and the argument above does not apply. The last example in Appendix C.1 shows that without convexity of $\mathcal{P}$ the improvement need not occur for small α.

D Data Experiments

All codes and experiments are on our Github page and will be made public soon.

D.1 Discussion on multinomial experiment

We highlight the following observations from Figure 2:

1. Focusing on the top row (without accumulation), when α_t is fixed, any increasing sequence n_t leads to consistent estimation, as evident from $R_t \to 0$. When α_t slowly decreases (second and third columns), the choice $n_t \asymp t^2$ (or $n_t \asymp t^3$) leads to consistent estimation, as suggested by our theoretical insights. Note that for $\alpha_t \asymp 1/\sqrt{t}$, it seems even the choice $n_t \asymp t$ leads to $R_t \to 0$. This suggests that while the condition in Proposition 3.4 is sufficient, it might not be necessary.
2. For the rapidly decaying $\alpha_t \asymp 1/t^2$ (the right-most column), R_t does not go to 0 for any of the choices of n_t considered here. However, we see that a faster rate $n_t \asymp t^3$ performs the best, and can get the limit R_∞ to be less than R_0. Although we do not achieve consistency in this case, iterative training is not bad in the sense that the model eventually performs better than the benchmark R_0 attained by training on samples coming exclusively from the ground-truth distribution.
3. Training with accumulation also has a similar trend. The major difference is visible when n_t is constant. Without accumulation, R_t becomes worse over iterations (similar to total collapse), while if the models are trained with accumulation, then it performs much better. For increasing n_t settings, the improvement is not so clear – it seems for such cases, training the model without accumulation might be better. For example, consider the third column, where $\alpha_t \asymp 1/t$. In this case, for both $n_t \asymp t^2$ and $n_t \asymp t^3$, the performance is better for the case without accumulation. With accumulation, it seems that R_t does not go to 0.

D.2 Additional numerical experiments

We consider the following four model classes, where the superscript $*$ indicates the true data distribution and the subscript t indicates the parameter estimated at iteration t:

1. Statistical Language Model in terms of Categorical Distribution: We set a fixed distribution over $K = 50$ classes, parametrized by $\theta^* \in \Delta^{K-1}$. We evaluate the fitted model in terms of $\hat{R}_t = \|\hat{\theta}_t - \theta^*\|_2^2$.
2. Gaussian model: We choose a fixed univariate Gaussian distribution $N(\mu_*, \sigma_*^2)$ as the true data distribution with $\mu^* = 0$ and $\sigma^* = 1$. We evaluate the fitted model in terms of $\hat{R}_t = W_2^2(N(\hat{\mu}_t, \hat{\sigma}_t^2), N(\mu_*, \sigma_*^2))$.
3. Gaussian Mixture Model (GMM): We choose a fixed univariate GMM with $K = 5$ components. Associating the mixing measure $G = \sum_k \pi_k \delta_{(\mu_k, \sigma_k^2)}$ with the GMM $\sum_k \pi_k N(\mu_k, \sigma_k^2)$, the performance of the fitted models is evaluated using the 1-Wasserstein distance between the mixing measures, i.e., $\hat{R}_t = W_1(\hat{G}_t, G^*)$.
4. Logistic Regression: We consider a simple binary classification task with features $X \sim N(0, I_2)$ (iid across all iterations and within an iteration). The true data generating model is $\log p(x)/(1 - p(x)) = X^\top \theta^*$, $\theta^* \in \mathbb{R}^3$ (where we include an intercept term) and $p(x) = P(y = 1|X = x)$. The fitted model is evaluated based on $\hat{R}_t = \|\hat{\theta}_t - \theta^*\|_2^2$.

Generic Simulation Details: For each of the above settings, we consider iterative training up to T rounds, where $T = 100$ for the first three model classes and $T = 50$ for logistic regression. Each trajectory for each setting (combination of model from above and the synthetic/fresh data generation scheme below) is repeated 500 times and the error plots are constructed along with a 95% standard error band. We consider 8 different settings based on how data is generated at each iteration. For each, at the first iteration the dataset is generated iid from the true model. We use $m_0 = 10$ (for multinomial and Gaussian experiments), $m_0 = 50$ (for GMM experiments) and $m_0 = 80$ (for logistic experiments), and denote $\alpha_0 = 0.3$ for all the settings. Recall m_t is the number of new previously

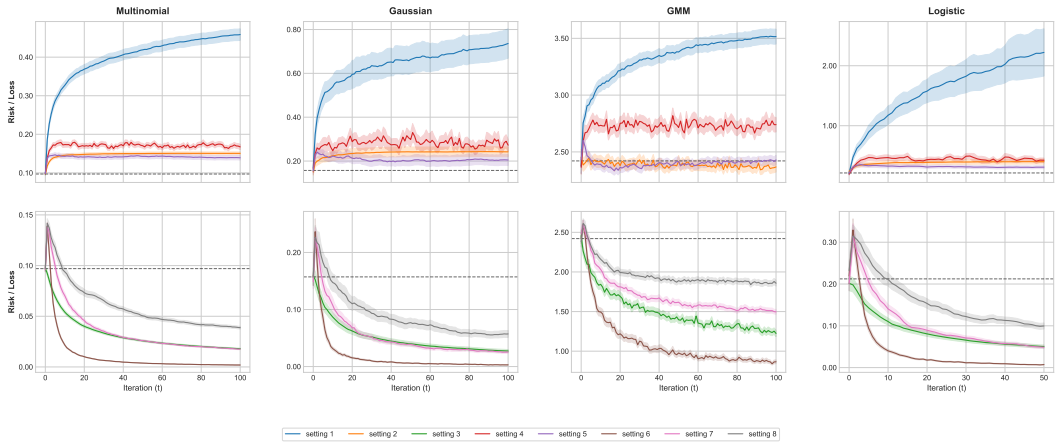


Figure 6: Simulation Studies: Iterative Evolution of iteratively Trained Statistical Models – Columns are model classes (categorical, Gaussian, GMM and logistic left to right) and the colors represent the different settings described above in the text. The two rows differentiate between settings where there is no iterative improvement (top) and there is noticeable iterative improvement (bottom). The horizontal dashed black line is the mean value of the risk at $t = 0$.

unseen sample introduced in iteration t, with the total size of training data at iteration t being n_t, given by $n_t = m_t$ (for settings without accumulation) and $n_t = \sum_{s \leq t} m_s$ (for settings with accumulation). Unless mentioned, there is *no accumulation* in the following settings.

1. Setting 1: This is the pure synthetic data regime with linearly increasing sample size – $m_t = m_0 t$ (where $n_0 = 10$ for categorical and Gaussian, $n_0 = 50$ for GMM and $n_0 = 80$ for logistic regression). In this setting $\alpha_t = 0$.
2. Setting 2: This is also pure synthetic data regime with quadratically increasing sample size $n_t = n_0 t^2$ – we consider this setting only for the categorical model owing to large computational time for other classes due to the huge amount of data.
3. Setting 3: This is the accumulation setting – we accumulate all the data from previous iterations for the current time point, generating n_0 new synthetic samples from the latest fitted model, thereby making $n_t = n_0 t$. Here again, $\alpha_t = 0$.
4. Setting 4: Fixed sample size and fixed fresh proportion – Use n_0 samples at each iteration drawn from a mixture model with $\alpha_t = \alpha_0$ fixed. We take $\alpha_0 = 0.3$.
5. Setting 5: Linearly increasing sample size $n_t = n_0 t$ and decreasing truth proportion $\alpha_t = \alpha_0 / t$.
6. Setting 6: Linearly increasing sample size $n_t = n_0 t$ and fixed truth proportion $\alpha_t = \alpha_0$, so that $n_t \alpha_t = \Omega(t)$.
7. Setting 7: Linearly increasing sample size $n_t = n_0 t$ and decreasing truth proportion $\alpha_t = \alpha_0 / t^{1/2}$ so that $n_t \alpha_t = \Omega(t^{1/2})$.
8. Setting 8: Linearly increasing sample size $n_t = n_0 t$ and decreasing truth proportion $\alpha_t = \alpha_0 / t^{2/3}$ so that $n_t \alpha_t = \Omega(t^{1/3})$.

The results from the experiments are shown in Figure 6. The left column shows the result for the categorical case, which we analyzed completely in Section 3 and complements the experiments in Section 5. The other columns show the results for the other models. Although the theoretical results in this work do not directly shed light on the behavior of these iterative evolution, we see that empirically, they behave very similar to the multinomial case, suggesting a wider applicability of our results. Firstly, in the pure synthetic data regime, we see that linearly growing n_t leads to total collapse, while quadratic increase (setting 2) leads to collapse, but the error is bounded. This is similar to the accumulation setting 3. In the case of fresh data available, we see settings 4 and 5 in the upper row, where $n_t \alpha_t$ is fixed (the limit in setting 4, given by our analysis in the previous section, is shown as the dashed horizontal line and matches the results). The last three settings, corresponding

to $n_t\alpha_t \uparrow \infty$, shown in the bottom row, all demonstrate that iterative training improves estimation even with synthetic data. Interestingly, in each model last 3 settings, the error increases a bit at the start and then starts decreasing.

D.3 Additional details for GPT-2 experiment

Here we describe the particular details of the network that we use for the simulations. At each iteration, we train the whole network from scratch (not just finetuning). We use `n_layer`=4 layers with `n_heads`=4 attention heads per layer, embedding size of `embd`=256, with context size `block_size`=128 and trained with Adam (learning rate 3×10^{-4}) optimizer using batch size `batch_size`=32. Overall, the model has about 16 million parameters.

Simulation settings: We consider 4 settings for these experiments and repeat each 10 times (each repetition involve randomly permuting the documents before the start of the train-validation split and the iterative process) over $T = 10$ iterative rounds. Despite running on a GPU (28 nodes: 8x NVIDIA A40 48GB), one complete experiment took around 30-40 minutes, where, surprisingly, generating from the model took slightly more time than training (over 3000 epochs). The 3000 epochs was used not just for time constraints, but we found that going beyond leads to sever over-fitting, where the training loss went down but the validation loss kept increasing.

E Filtering via a real-vs-synthetic classifier

At stage t, let $\mathcal{C}_t$ be a classifier trained to distinguish real samples $Y \sim P^*$ from synthetic samples $Y \sim \widehat{P}_t$ (the latest model). We assume $\mathcal{C}_t$ is applied to each observation in the next-stage training batch $Y^{t+1} = \{Y_1^{t+1}, \ldots, Y_{n_{t+1}}^{t+1}\}$, and we retain only those classified as real, i.e. $\mathcal{C}_t(Y_i^{t+1}) = 1$. Denote the retained (filtered) batch by $\tilde{Y}^{t+1}$, with random size $\tilde{n}_{t+1} := |\tilde{Y}^{t+1}|$.

Recall that we assume the classifier $\mathcal{C}_t$ has type-I error rate e_{1t} and type-II error rate e_{2t} with the null corresponding to P^* and alternate corresponding to $\hat{P}_t$, i.e.

$$e_{1t} := \mathbb{P}\big(\mathcal{C}_t(Y) = 0 \mid Y \sim P^*\big), \qquad e_{2t} := \mathbb{P}\big(\mathcal{C}_t(Y) = 1 \mid Y \sim \widehat{P}_t\big).$$

Then each point is retained with probability

$$q_{t+1} := \mathbb{P}\big(\mathcal{C}_t(Y) = 1\big) = \alpha_{t+1}(1 - e_{1t}) + (1 - \alpha_{t+1})e_{2t},$$

so $\tilde{n}_{t+1} \sim \mathrm{Binomial}(n_{t+1}, q_{t+1})$ and hence

$$\mathbb{E}[\tilde{n}_{t+1}] = q_{t+1}\, n_{t+1} = \big[\alpha_{t+1}(1 - e_{1t}) + (1 - \alpha_{t+1})e_{2t}\big]\, n_{t+1}.$$

Moreover, the filtered data $\tilde{Y}^{t+1}$ are i.i.d. from a mixture distribution

$$\tilde{P}_{t+1} = \tilde{\alpha}_{t+1}P^* + (1 - \tilde{\alpha}_{t+1})\widehat{P}_t, \qquad \tilde{\alpha}_{t+1} = \frac{\alpha_{t+1}(1 - e_{1t})}{\alpha_{t+1}(1 - e_{1t}) + (1 - \alpha_{t+1})e_{2t}}.$$

Define the filtered estimator $\tilde{\theta}_t$ as the empirical class-frequency vector based on the filtered data $\tilde{Y}^t$:

$$\tilde{\theta}_t(k) := \frac{1}{\tilde{n}_t} \sum_{j=1}^{\tilde{n}_t} \mathbf{1}\{\tilde{Y}_j^t = k\}, \qquad k \in \{1, \ldots, K\},$$

(with the convention that the procedure resamples until $\tilde{n}_t \geq 1$, or equivalently all statements below are conditional on $\{\tilde{n}_t \geq 1\}$). We measure performance via the filtered risk $R_t^f := \mathbb{E}\big\|\tilde{\theta}_t - \theta^*\big\|_2^2$, where the expectation is taken over randomness of both data Y^t and induced by the filtering/classifier decisions (equivalently the induced random retained set / retained sample size).

Assumption E.1 (Label-agnostic filtering)**.** For each stage $t \geq 0$, the classifier $\mathcal{C}_t$ is applied to each observation Y drawn from the mixture $\alpha_{t+1}P^* + (1 - \alpha_{t+1})\widehat{P}_t$. Conditional on whether Y is drawn from P^* or from $\widehat{P}_t$, the classification decision of $\mathcal{C}_t$ is independent of the categorical label of Y. Equivalently, for any class k,

$$\mathbb{P}\big(\mathcal{C}_t(Y) = 1 \mid Y \sim P^*,\, Y = k\big) = \mathbb{P}\big(\mathcal{C}_t(Y) = 1 \mid Y \sim P^*\big)$$
$$\mathbb{P}\big(\mathcal{C}_t(Y) = 1 \mid Y \sim \widehat{P}_t,\, Y = k\big) = \mathbb{P}\big(\mathcal{C}_t(Y) = 1 \mid Y \sim \widehat{P}_t\big),$$

where these probabilities equal $1 - e_{1t}$ and e_{2t}, respectively.

Theorem E.2 (Filtered-risk recurrence)**.** *Let $\tilde{R}_t := \mathbb{E}\|\tilde{\theta}_t - \theta^*\|_2^2$. Under the filtration model described above satisfying Assumption E.1, with error rates e_{1t}, e_{2t} that are deterministic (i.e., do not depend on the current fit), for each $t \geq 1$,*

$$\tilde{R}_t = \mathbb{E}\Big[\frac{1}{\tilde{n}_t}\Big]\,(1 - \|\theta^*\|_2^2) + \Big(1 - \mathbb{E}\Big[\frac{1}{\tilde{n}_t}\Big]\Big)\,(1-\tilde{\alpha}_t)^2\,\tilde{R}_{t-1}, \tag{2}$$

where $\mathbb{E}[1/\tilde{n}_t]$ is understood under the same convention used to define $\tilde{\theta}_t$ (i.e. conditioning on $\{\tilde{n}_t \geq 1\}$).

Proof. Fix $t \geq 1$ and write

$$\rho_t := \tilde{\alpha}_t\,\theta^* + (1-\tilde{\alpha}_t)\,\tilde{\theta}_{t-1}.$$

Conditional on $(\tilde{\theta}_{t-1}, \tilde{n}_t)$ (with $\tilde{n}_t \geq 1$), the filtered batch $\tilde{Y}^t$ is i.i.d. $\mathrm{Cat}(\rho_t)$ and $\tilde{\theta}_t$ is the empirical frequency vector based on $\tilde{n}_t$ samples. Hence,

$$\mathbb{E}\Big[\|\tilde{\theta}_t - \theta^*\|_2^2 \,\Big|\, \tilde{\theta}_{t-1}, \tilde{n}_t\Big] = \underbrace{\mathbb{E}\Big[\|\tilde{\theta}_t - \rho_t\|_2^2 \,\Big|\, \tilde{\theta}_{t-1}, \tilde{n}_t\Big]}_{\text{variance}} + \underbrace{\|\rho_t - \theta^*\|_2^2}_{\text{bias}^2}. \tag{3}$$

The two terms are explicit:

$$\mathbb{E}\Big[\|\tilde{\theta}_t - \rho_t\|_2^2 \,\Big|\, \tilde{\theta}_{t-1}, \tilde{n}_t\Big] = \frac{1 - \|\rho_t\|_2^2}{\tilde{n}_t}, \qquad \|\rho_t - \theta^*\|_2^2 = (1-\tilde{\alpha}_t)^2\|\tilde{\theta}_{t-1} - \theta^*\|_2^2.$$

Therefore,

$$\mathbb{E}\Big[\|\tilde{\theta}_t - \theta^*\|_2^2 \,\Big|\, \tilde{\theta}_{t-1}, \tilde{n}_t\Big] = \frac{1 - \|\rho_t\|_2^2}{\tilde{n}_t} + (1-\tilde{\alpha}_t)^2\|\tilde{\theta}_{t-1} - \theta^*\|_2^2. \tag{4}$$

Unbiasedness of $\tilde{\theta}_t$. We first verify directly that $\mathbb{E}\tilde{\theta}_1 = \theta^*$. Conditional on $\widehat{\theta}_0$, the filtered sample $\tilde{Y}^1$ is i.i.d. from

$$\tilde{P}_1 = \tilde{\alpha}_1 P^* + (1-\tilde{\alpha}_1)\widehat{P}_0 = \mathrm{Cat}\Big(\tilde{\alpha}_1\theta^* + (1-\tilde{\alpha}_1)\widehat{\theta}_0\Big),$$

so, for each coordinate k (and conditioning on $\tilde{n}_1 \geq 1$),

$$\mathbb{E}\big[\tilde{\theta}_1(k) \mid \widehat{\theta}_0\big] = \tilde{\alpha}_1\theta^*(k) + (1-\tilde{\alpha}_1)\widehat{\theta}_0(k).$$

Taking expectation and using $\mathbb{E}\widehat{\theta}_0 = \theta^*$ yields

$$\mathbb{E}\tilde{\theta}_1(k) = \tilde{\alpha}_1\theta^*(k) + (1-\tilde{\alpha}_1)\mathbb{E}\widehat{\theta}_0(k) = \theta^*(k).$$

Thus $\mathbb{E}\tilde{\theta}_1 = \theta^*$. For the induction step, assume $\mathbb{E}\tilde{\theta}_{t-1} = \theta^*$. By the same reasoning,

$$\mathbb{E}\big[\tilde{\theta}_t \mid \tilde{\theta}_{t-1}\big] = \tilde{\alpha}_t\theta^* + (1-\tilde{\alpha}_t)\tilde{\theta}_{t-1},$$

so taking expectations gives

$$\mathbb{E}\tilde{\theta}_t = \tilde{\alpha}_t\theta^* + (1-\tilde{\alpha}_t)\mathbb{E}\tilde{\theta}_{t-1} = \theta^*.$$

Next, by the label-agnostic filtering assumption, the random count $\tilde{n}_t$ depends only on the keep probabilities (equivalently on $(\alpha_t, e_{1,t-1}, e_{2,t-1})$) and not on the categorical labels; in particular, since $e_{1,t-1}, e_{2,t-1}$ are deterministic, $\tilde{n}_t$ is independent of $\tilde{\theta}_{t-1}$, and hence independent of ρ_t (which is a function of $\tilde{\theta}_{t-1}$). Taking expectation of (4) and using this independence yields

$$\tilde{R}_t = (1-\tilde{\alpha}_t)^2\tilde{R}_{t-1} + \mathbb{E}\Big[\frac{1}{\tilde{n}_t}\Big]\,\mathbb{E}\big[1 - \|\rho_t\|_2^2\big]. \tag{5}$$

Finally, expand $\|\rho_t\|_2^2$ and use $\mathbb{E}\tilde{\theta}_{t-1} = \theta^*$:

$$\|\rho_t\|_2^2 = \|\theta^*\|_2^2 + (1-\tilde{\alpha}_t)^2\|\tilde{\theta}_{t-1} - \theta^*\|_2^2 + 2(1-\tilde{\alpha}_t)\theta^{*\top}(\tilde{\theta}_{t-1} - \theta^*),$$

so after taking expectations the cross term vanishes and

$$\mathbb{E}\|\rho_t\|_2^2 = \|\theta^*\|_2^2 + (1-\tilde{\alpha}_t)^2\tilde{R}_{t-1}.$$

Hence

$$\mathbb{E}\big[1-\|\rho_t\|_2^2\big] = 1-\|\theta^*\|_2^2-(1-\tilde{\alpha}_t)^2\tilde{R}_{t-1}.$$

Substitute this into (5) and rearrange to obtain (2):

$$\tilde{R}_t = (1-\tilde{\alpha}_t)^2\tilde{R}_{t-1}+\mathbb{E}\Big[\frac{1}{\tilde{n}_t}\Big]\Big(1-\|\theta^*\|_2^2-(1-\tilde{\alpha}_t)^2\tilde{R}_{t-1}\Big) = \mathbb{E}\Big[\frac{1}{\tilde{n}_t}\Big](1-\|\theta^*\|_2^2)+\Big(1-\mathbb{E}\Big[\frac{1}{\tilde{n}_t}\Big]\Big)(1-\tilde{\alpha}_t)^2\tilde{R}_{t-1}.$$

□

Remark E.3 (Mean-field effective-sample-size approximation). A convenient closed-form approximation replaces $\tilde{n}_{t+1}$ by its mean $\mathbb{E}\tilde{n}_{t+1} = q_{t+1}n_{t+1}$, yielding

$$\tilde{R}_{t+1} \approx (1-\tilde{\alpha}_{t+1})^2\tilde{R}_t + \frac{1-\|\theta^*\|_2^2-(1-\tilde{\alpha}_{t+1})^2\tilde{R}_t}{q_{t+1}n_{t+1}}, \qquad q_{t+1} = \alpha_{t+1}(1-e_{1t})+(1-\alpha_{t+1})e_{2t}.$$

This highlights the tradeoff induced by filtering: increasing $\tilde{\alpha}_{t+1}$ improves the "fresh information" fraction but decreases the effective sample size through q_{t+1}.

Remark E.4 (Adaptive classifiers). For adaptive classifiers whose error rates depend on the current fit, such as the oracle classifier of Section 3, for which $e_{2,t-1}$ depends on $\|\hat{\theta}_{t-1}-\theta^*\|_1$, both $\tilde{\alpha}_t$ and $\tilde{n}_t$ are functions of $\tilde{\theta}_{t-1}$. Then $\tilde{n}_t$ and ρ_t are dependent, and the factorization leading to (5) fails. The filtered data is still an exact reweighting at each step, i.e., conditionally on $\tilde{\theta}_{t-1}$ the batch $\tilde{Y}^t$ is i.i.d. from $\tilde{\alpha}_t P^* + (1-\tilde{\alpha}_t)\widehat{P}_{t-1}$, and the conditional one-step identity (4) continues to hold. However, Theorem E.2 does not give a closed-form trajectory of $\tilde{R}_t$ in this case, and we do not claim one.

F Naturally Adaptive Mixture Weights α_t

Here we present details and additional examples of adaptive schemes of mixture weights α_t, hinted in our discussion in Section 6.

For example, if the mixture weights α_t follow either of the following adaptive conditions for all $t \geq 1$:

(i) $(1-\alpha_t)^2 \leq \min\Big\{1, \frac{1}{\lambda R_{t-1}}\Big\}$, for some constant $\lambda > 0$;

(ii) $(1-\alpha_t)^2 \leq \exp(-\lambda R_{t-1})$, for some $\lambda > 1$.

then model collapse is avoided. Moreover, optimistic upper-bounds for the limiting risk can be obtained, suggesting natural improvement and limiting consistency of the iterative training process.

Below we provide the general result from which the above remarks follow.

Proposition F.1. *Let $g_\lambda(.) : [0,2] \to (0,1]$ be any decreasing function with constant $\lambda > 1$, such that for all $x \in [0,2]$, $g_\lambda(x) < \frac{1}{\lambda x}$. Then collapse is avoided if $(1-\alpha_t)^2 \leq g_\lambda(R_{t-1})$ for all $t \geq 1$ and the following holds:*

1. *For fixed sample size $n_t = n$ (with $n_0 = n$),*

$$R_t \leq R_0 + \frac{n-1}{\lambda n};$$

2. *For any sequence $n_t \uparrow \infty$,*

$$R_\infty \leq \frac{1}{\lambda}.$$

Proof. Recall

$$\begin{aligned} R_t &= \frac{n_0 R_0}{n_t} + \frac{n_t-1}{n_t}(1-\alpha_t)^2 R_{t-1} \\ &\leq \frac{n_0 R_0}{n_t} + \frac{n_t-1}{n_t} g_\lambda(R_{t-1}) R_{t-1} \\ &< \frac{n_0 R_0}{n_t} + \frac{n_t-1}{n_t}\cdot\frac{1}{\lambda}\left[\text{as } g_\lambda(x) < \frac{1}{\lambda x}\right]. \end{aligned}$$

Replacing $n_t = n$ and $n_0 = n$, we directly get

$$R_t \leq R_0 + \frac{n-1}{\lambda n},$$

and for $n_t \uparrow \infty$, the first term vanishes and the ratio $\frac{n_t - 1}{n_t}$ converges to 1 giving

$$R_\infty \leq \frac{1}{\lambda}.$$

□

G Detailed comparison with existing work

Table 1 summarizes, for each closely related work, the data-generating setting, the quantity studied, and what is new in the present paper. A detailed discussion follows.

Work	Data-generating setting	Quantity studied	What is new here
[43, 44]	Fully synthetic retraining ($\alpha_t = 0$, $n_t = m_t$); Gaussian and discrete models	Variance collapse, distributional drift	Fresh real data with evolving α_t and n_t; conditions for improvement and consistency
[1]	Self-consuming loops with fixed or fresh real data (empirical)	Sample quality and diversity across generations	Exact risk trajectory (Theorem 3.1) and precise (α_t, n_t) conditions
[16]	Regression; purely synthetic, previous models label new data	Test error, scaling laws	Generative setting with fresh real data from a mixture
[5]	Fixed real sample D_0 reused every round plus synthetic samples; real/synthetic parts known to the learner	Stability of iterative retraining under a locality assumption	Fresh, previously unseen real data; source-agnostic training; improvement and consistency
[42]	Categorical next-token model; fully synthetic, or synthetic mixed with a fixed original sample; no accumulation	Upper bounds on the synthetic proportion avoiding collapse	Evolving population-level mixture with fresh data, accumulation, exact trajectory; we inherit only the categorical abstraction
[21]	One-hidden-layer diffusion model; mixture of true and previously fitted models; no accumulation	$\mathsf{TV}(\hat{P}_t, P^*)$ bounds	Exact trajectory; growing n_t with possibly vanishing α_t; accumulation
[23]	Accumulation of real and synthetic data; linear regression; no fresh real data	Test error, bounded by a $\pi^2/6$ multiple	Accumulation combined with fresh data, which yields consistency
[30]	Accumulation; empirical across model classes, theory for univariate Gaussian	Rate of performance degradation	As above
[13]	Accumulation; general M-estimation; no fresh real data	Test risk ($\pi^2/6$ universality)	As above
[3]	Source-agnostic MLE on accumulated data, $m_t = m$, $\alpha_t = 0$	KL/TV stays bounded	With fresh data the risk can decrease below R_0 or vanish
[33]	Temperature-controlled generative models; accumulation; no fresh data; martingale-difference assumption	Long-run behavior of the learned distribution	Fresh data; exact trajectory without martingale assumptions
This paper	Categorical next-token model; unlabeled mixture of fresh real (α_t, possibly vanishing) and synthetic data; with or without accumulation; varying n_t; source-agnostic MLE	Exact trajectory of $R_t = \mathbb{E}\|\hat{\theta}_t - \theta^*\|_2^2$; collapse vs. improvement vs. consistency	Also general necessity/sufficiency of fresh data beyond the categorical model (Section 4)

Table 1: Comparison with closely related work.

Model collapse was first studied in [44], where the phenomenon of variance collapse was illustrated in univariate Gaussian. This drew attention as [1] demonstrated that many datasets are already contaminated with synthetic samples. There is a series of works reporting some level of performance degradation in models trained on such data [44, 1, 27, 6, 7, 25]. Among theoretical works, several types of synthetic data contamination settings have been considered in various model classes. [43, 44] consider a fully synthetic (in our notations, $n_t = m_t$ and $\alpha_t = 0$) regime with Gaussian and discrete distributions, showing variance explosion. [16] tackles the regression setting, under purely synthetic regime and extended to neural scaling laws [17, 18]. These works mostly focus on discriminative models, where previous models are used to label new data. Moreover, the setting of [17] is non-iterative.

Next, we move to papers that consider some form of mixture structure (or other ways to inject fresh information), going beyond simple pure synthetic settings. [5] deals with a general class of maximum likelihood estimators, under a setting where at iteration t, the training data is $D_t = D_0 \cup \tilde{D}_t$,

where $\tilde{D}_t$ consists of synthetic samples generated from the last-fitted model, and D_0 is the fixed collection of samples from the true distribution, already used at $t = 0$. Thus, despite its mixture-like appearance, this setting does not consider fresh injection of previously unseen samples from the true distribution. Further, their formulation of iterative retraining (definition of $\mathcal{G}_\lambda(\theta)$ or Θ^n_{t+1}) shows that it assumes knowledge of which part is the real, and which part is synthetic – this is different from our setting where model training at each iteration is assumed agnostic to the source of the samples. [42] works with categorical distributions without accumulation, with either fully synthetic structure, or a partially synthetic structure – comprised of synthetic samples from last-fitted model and first fitted-model (which they call real data, since they assume access to only a fixed sample from the true distribution). Our work is motivated by their setting, however, we analyze a much more realistic setting involving accumulation and mixture, and identify the nuanced balance of n_t and α_t that aids iterative improvement or consistency. [20] discuss avenues to prevent collapse with human feedback loops – such feedback is a different notion of fresh information injected into the system. [21] analyzed model collapse in a simplified one-hidden-layer diffusion model without accumulation, allowing the training data to arise from a mixture of true distribution and previously fitted models. On the other hand, [22] analyzed several architectures under a framework they called recursive stability, which bears similarities to algorithmic stability.

Another prominent line of work highlights the role of accumulation to prevent model collapse [30, 23]. [30] provides empirical verification that with accumulation, the degradation of performance of the iterative estimator is much slower compared to without accumulation, in various classes of models from Gaussian models, to kernel density estimators and supervised fine-tuning of language models, and provides justification in the univariate Gaussian case. [23] analyzes the benefit of accumulation in the linear regression setting, showing that the test error stays bounded. In particular, test risk at any later generation is bounded by a moderate multiple, viz. $\pi^2/6$ of the test risk of training with the original real data alone – a phenomenon later explored in a more general M-estimation setting [13]. The recent work in [3] demonstrates collapse can be avoided through source-agnostic MLE on accumulated data (with $m_t = m$ fixed, and $\alpha_t = 0$ in our notations) under general conditions – however, since there is no injection of fresh data from the ground truth, the paper shows that KL or TV stays bounded, but the upper bound does not go to 0 over iterations. In [33], the authors analyze model collapse for a class of generative models, controlled via a temperature parameter, with accumulation and without fresh data injection, but rely on assuming that the differences between distributions in subsequent generations form a martingale difference sequence. For a broader discussion regarding the different settings and definitions (often contradictory) used in this vast literature of model collapse, the interested reader is recommended to look into [40].

While our theoretical analysis focuses on a tractable model class, it provides a remarkably precise handle on risk dynamics within a setting of unprecedented generality. Specifically, our framework accommodates an unlabeled streaming mixture of ground-truth and synthetic data, data accumulation, iteration-varying sample sizes (m_t or n_t), and evolving fresh data proportions (α_t), all while keeping training agnostic to the source of the samples. This exact characterization reveals the subtle interplay between dataset growth and real-data proportion that governs the phase transition between collapse, iterative improvement and consistency – uncovering several key phenomena unexplained by prior literature. For instance, without data accumulation, even a slowly decaying α_t ($\sum_t \alpha_t = \infty$, such as $\alpha_t = 1/\sqrt{t}$) can achieve consistency provided the sample size scales as $n_t \asymp t^2$. On the other hand, under data accumulation, a constant size of new samples ($m_t = m$) with a fixed $\alpha > 0$ guarantees consistency, requiring no non-trivial lower bound on α. Extensive simulation studies demonstrate that these theoretical mechanisms extend well beyond our tractable model class to broader families, including Gaussian mixture models and logistic regression. Ultimately, this work provides a rigorous foundation for extending such nuanced risk analyses to complex model classes, and highlights the promise of developing mixture-aware training objective over standard source-agnostic MLE.

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: The abstract and Section 1 state the contributions, which match the technical results in Sections 3–5

   Guidelines:

   - The answer [N/A] 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 [N/A] 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: Limitations are discussed in the text following Proposition 3.4 (sample-size schedule depends on the typically unknown α_t) and in Section 6 (assumption of a fixed true distribution, difficulty of estimating mixture weights, restriction to mixture-based modeling of fresh information).

   Guidelines:

   - The answer [N/A] 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: All theorems, corollaries, and propositions are stated with their assumptions in Sections 3 and 4, with complete proofs in Appendices B, C, E, and F.

Guidelines:

- The answer [N/A] 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: Section 5 specifies the multinomial simulation settings (schedules of α_t, n_t, K, T, 100 repetitions, MLE) and the public dataset, and Appendix D.3 gives full nanoGPT hyperparameters.

Guidelines:

- The answer [N/A] 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: [No]

   Justification: Code is not yet publicly released, but the experiments use a public dataset (Wikipedia movie plots on Kaggle) and the open-source nanoGPT, with full settings described in Section 5 and Appendix D.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.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://neurips.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) necessary to understand the results?

   Answer: [Yes]

   Justification: Section 5 lists all simulation parameters and the 2000-document held-out split, and Appendix D.3 specifies the nanoGPT architecture, optimizer, learning rate, batch size, and epoch count.

   Guidelines:

   - The answer [N/A] 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: The multinomial experiments in Figures 2 and 3 are repeated 100 times, the simulations in Figure 6 are repeated 500 times, and the GPT-2 experiments in Figure 4 are repeated 10 times; these figures show the mean curve with a standard-error band across the repetitions.

   Guidelines:

- The answer [N/A] 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: Appendix D.3 reports the GPT-2 experiments were run on a GPU cluster (28 nodes, each 8×NVIDIA A40 48GB) with a wall-clock time of 30–40 minutes per complete experiment; the other simulations are lightweight and run on standard CPUs.

   Guidelines:

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

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: The work is theoretical, uses only publicly available data and open-source code, and conforms with the NeurIPS Code of Ethics.

   Guidelines:

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

10. **Broader impacts**

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

Answer: [Yes]

Justification: Section 1 and Section 6 discuss both the positive impact of understanding when iterative training avoids collapse and the negative implications of uncontrolled training on AI-generated content.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] 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., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: The paper releases no pre-trained model, generator, or scraped dataset, so no safeguards are required.

Guidelines:

- The answer [N/A] 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: The Wikipedia movie plots dataset is cited via its Kaggle URL in Section 5 and the nanoGPT codebase is cited (ref. [29]); both are used under their original public terms.

Guidelines:

- The answer [N/A] 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: [N/A]

Justification: The paper does not release any new datasets, models, or other assets.

Guidelines:

- The answer [N/A] 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: [N/A]

Justification: The paper does not involve crowdsourcing or human subjects.

Guidelines:

- The answer [N/A] 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: [N/A]

Justification: The paper does not involve human subjects, so no IRB review is required.

Guidelines:

- The answer [N/A] 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 rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: LLMs are an object of study (a small GPT-2 in Section 5 / Appendix D.3) but are not used as a non-standard component of the core methodology.

Guidelines:

- The answer [N/A] 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 in the NeurIPS handbook for what should or should not be described.