Accumulating Data Avoids Model Collapse

Anonymous Author(s)
Affiliation
Address
email

Abstract

The increasing prevalence of AI-generated content on the internet raises a critical and timely question: What happens when generative machine learning models are pretrained on web-scale datasets containing data created by earlier generative models? Recent studies have highlighted a phenomenon termed "model collapse," whereby model performance degrades over iterations, rendering newer generative models unusable. However, other recent research questioned the practical relevance of model collapse by providing evidence that (1) model collapse was caused by deleting past data en masse and then training largely (or entirely) on purely synthetic data from the latest generative model, and (2) model collapse is avoided if new synthetic data are instead added to existing real and synthetic data. These two claims are particularly important in forecasting likely futures of deep generative models pretrained on web-scale data because, in practice, web-scale data is not deleted en masse and new synthetic data accumulates alongside existing real and synthetic data. In this work, we test whether these two claims hold on three new prominent settings for studying model collapse: multivariate Gaussian modeling, supervised finetuning of language models and kernel density estimation. In all three of the new settings, we find that the two claims hold: model collapse is indeed caused by deleting past data en masse, and model collapse is avoided by accumulating new synthetic data alongside past data.

1 Introduction

With each day, the internet contains increasingly more AI-generated content[1]. What does this observation imply for the future of deep generative models pretrained on web-scale datasets containing data generated by their predecessors? Previous work forewarned that such model-data feedback loops exhibit *model collapse*, a phenomenon whereby model performance degrades with each model-fitting iteration such that newer models trend towards useless [12, 15, 19, 2, 16, 4, 3, 5, 8, 9, 14, 11]. However, more recent work has challenged this narrative [10, 18, 14]. Of particular interest to us is Gerstgrasser et al. [10], which made two claims:

1. Many previous model collapse papers induced model collapse by deleting past data en masse and training largely (or solely) on synthetic data from the latest generative model, and
2. If new synthetic data are instead added to real data, i.e., data accumulate over time, then model collapse is avoided.

These two claims are relevant to forecasting the future of deep generative models because, if correct, model collapse is significantly less likely to pose a realistic threat since accumulating data over time is a more faithful model of reality; as a partner at Andreessen Horowitz elegantly explained, deleting data en masse is "not what is happening on the internet. We won't replace the Mona Lisa or Lord of

[1] Tweet by Sam Altman on Feb 9th, 2024

Submitted to the Mathematics of Modern Machine Learning (M3L) Workshop at NeurIPS 2024.

the Rings with AI generated data, but the classics will continue to be part of the training data set and exist along with it."[2]. We emphasize that when discussing deleting past data *en masse*, we mean that (almost) *all* previous data are deleted. In the context of pretraining on web-scale data, the correct mental picture is that the entirety of the internet is deleted, not that a single minor website disappears.

However, a recent prominent paper [20] introduced three new settings for studying model collapse that were not studied by Gerstgrasser et al. [10]. The three new settings are:

1. **Multivariate Gaussian Modeling:** Multivariate Gaussians are repeatedly fit to data and then used to sample new synthetic data for future Gaussian fitting.
2. **Supervised Finetuning of Language Models:** Language models are finetuned in a supervised manner and then used to sample new synthetic text for future finetuning.
3. **Kernel Density Estimation:** Kernel density estimators are repeatedly fit to data and then used to sample new synthetic data for future kernel density estimators.

In this work, we ask whether the two model collapse claims hold in these three new settings. We find both claims do. In multivariate Gaussian modeling, we find that model collapse is caused by deleting past data en masse, and mitigated by instead accumulating synthetic data with previous real and synthetic data. In supervised finetuning of language models and kernel density estimation, we again find consistent results. The consistency of these results across different model types and datasets suggests that *this distinction is a general phenomenon, and is not specific to any particular model or dataset or learning algorithm.*

Interestingly, we discover in kernel density estimation that training on real and accumulating synthetic data can yield *lower loss on real test data* than training on real data alone. This result matches the language model pretraining results of Gerstgrasser et al. [10], but is significantly faster to experiment with and significantly easier to study mathematically. We leave answering the questions of under what conditions, and why, synthetic data can lead to lower loss on real test data to future work.

2 Model Collapse in Multivariate Gaussian Modeling

Recent prominent work [20] introduced a simplified setting for studying model collapse: repeatedly fitting multivariate Gaussians to data and sampling from the fit Gaussians. In this setting, one begins with n *real* data drawn from a multivariate Gaussian with mean $\mu^{(0)}$ and covariance $\Sigma^{(0)}$:

$$X_1^{(0)}, ..., X_n^{(0)} \sim_{i.i.d.} \mathcal{N}(\mu^{(0)}, \Sigma^{(0)}).$$

To study model-data feedback loops, we alternate two stages: model-fitting and sampling. For model fitting, one computes the unbiased mean and covariance of the most recent data:

$$\hat{\mu}_{\text{Replace}}^{(t+1)} \stackrel{\text{def}}{=} \frac{1}{n} \sum_{j=1}^{n} X_j^{(t)} \tag{1}$$

$$\hat{\Sigma}_{\text{Replace}}^{(t+1)} \stackrel{\text{def}}{=} \frac{1}{n-1} \sum_{j=1}^{n} (X_j^{(t)} - \hat{\mu}_{\text{Replace}}^{(t+1)})(X_j^{(t)} - \hat{\mu}_{\text{Replace}}^{(t+1)})^T \tag{2}$$

For model sampling, one samples m new synthetic data using the fit Gaussian parameters:

$$X_1^{(t)}, ..., X_n^{(t)} \,\Big|\, \hat{\mu}_{\text{Replace}}^{(t)}, \hat{\Sigma}_{\text{Replace}}^{(t)} \quad \sim_{i.i.d.} \quad \mathcal{N}(\hat{\mu}_{\text{Replace}}^{(t)}, \hat{\Sigma}_{\text{Replace}}^{(t)}).$$

Under the above data-model feedback loop, Shumailov et al. [20] prove that

$$\hat{\Sigma}_{\text{Replace}}^{(t+1)} \stackrel{a.s.}{\to} 0 \quad ; \quad \mathbb{E}[\mathbb{W}_2^2(\mathcal{N}(\hat{\mu}_{\text{Replace}}^{(t+1)}, \hat{\Sigma}_{\text{Replace}}^{(t+1)}), \mathcal{N}(\mu^{(0)}, \Sigma^{(0)}))] \to \infty \text{ as } t \to \infty,$$

where $\mathbb{W}_2$ denotes the Wasserstein-2 distance. This result states that the fit covariance will collapse to 0 and that the Wasserstein-2 distance will diverge as this model-data feedback loop unfolds[3].

[2] LinkedIn Post by Guido Appenzeller on July 28th, 2024.

[3] Note: the Wasserstein-2 distance diverges not because the covariance collapses to 0 but because the distance between the t-th fit mean $\hat{\mu}_{\text{Replace}}^{(t)}$ and the true mean $\mu^{(0)}$ diverges.

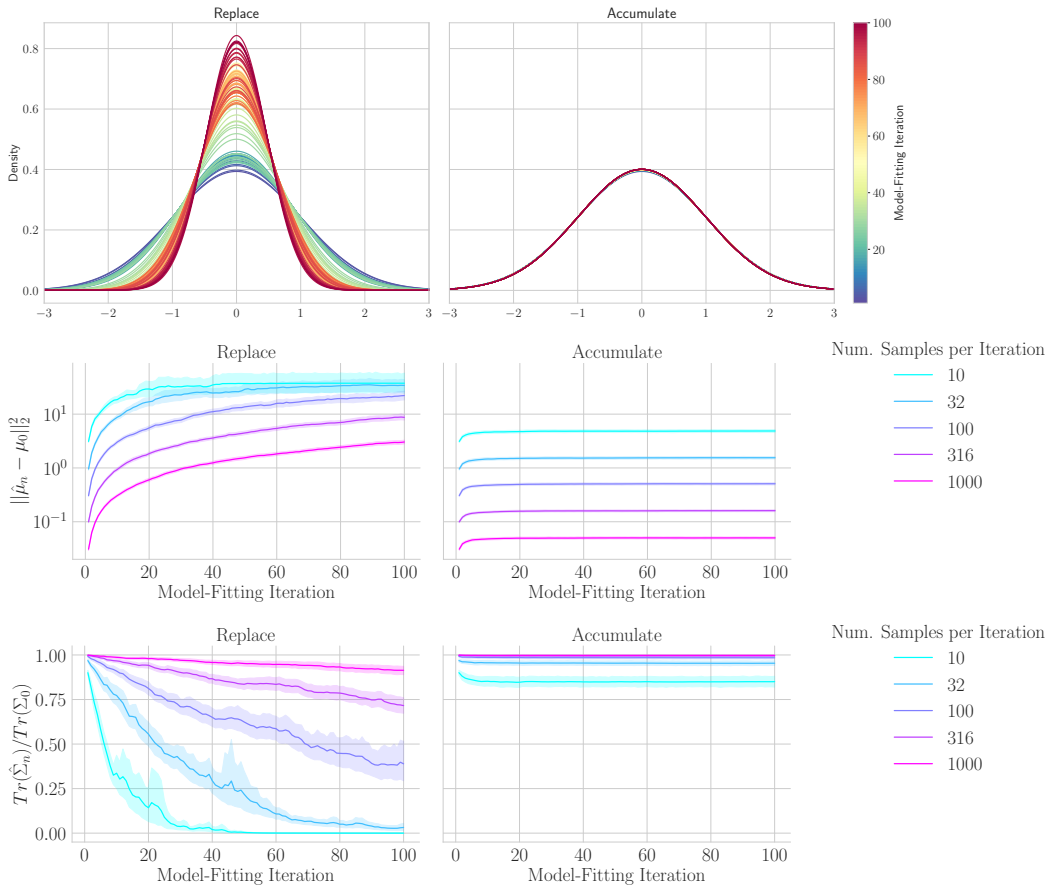

Figure 1: **Model Collapse in Multivariate Gaussian Modeling. Top:** Previous work [20] proves model collapse occurs if one iteratively fits means and covariances to data and then samples new data from a Gaussian with the fitted parameters (left). We demonstrate that if one doesn't delete all data after each model-fitting iteration - i.e., if data accumulate - then model collapse does not occur (right). Number of Samples Per Iteration: 316. Note: We visualize the fit Gaussians as zero-mean for easy comparison of the fit covariances across model-fitting iterations. **Middle:** If data are replaced, then the empirically fit means drift away from the original data's mean with increasing model-fitting iterations, but if data instead accumulate, then the empirically fit means stabilize. **Bottom:** If data are replaced, then the empirically fit covariances collapse compared to the original data's covariance, but if past data are not discarded, then the fit covariances solidify quickly and collapse is avoided.

However, *this result assumes that all data are deleted after each model-fitting iteration*. As discussed in Sec. 1, we consider this assumption unrealistic. Following Gerstgrasser et al. [10], we instead ask: what happens if data instead *accumulate* across model-fitting iterations? To study this, we instead consider Gaussian parameters fit using data across *all* $t+1$ iterations with n samples per iteration:

$$\hat{\mu}^{(t+1)}_{\text{Accumulate}} \stackrel{\text{def}}{=} \frac{1}{n(t+1)} \sum_{i=0}^{t} \sum_{j=1}^{n} X_j^{(i)} \tag{3}$$

$$\hat{\Sigma}^{(t+1)}_{\text{Accumulate}} \stackrel{\text{def}}{=} \frac{1}{n(t+1)-1} \sum_{i=0}^{t} \sum_{j=1}^{n} (X_j^{(i)} - \hat{\mu}^{(t+1)}_{\text{Accumulate}})(X_j^{(i)} - \hat{\mu}^{(t+1)}_{\text{Accumulate}})^T \tag{4}$$

Data are then sampled using these fit accumulation parameters $\hat{\mu}^{(t)}_{\text{Accumulate}}, \hat{\Sigma}^{(t)}_{\text{Accumulate}}$ rather then the fit replacement parameters $\hat{\mu}^{(t)}_{\text{Replace}}, \hat{\Sigma}^{(t)}_{\text{Replace}}$.

Empirically, we find that deleting all data after each model-fitting iteration causes model collapse (Fig. 1, left), whereas accumulating data across model-fitting iterations prevents model collapse (Fig.

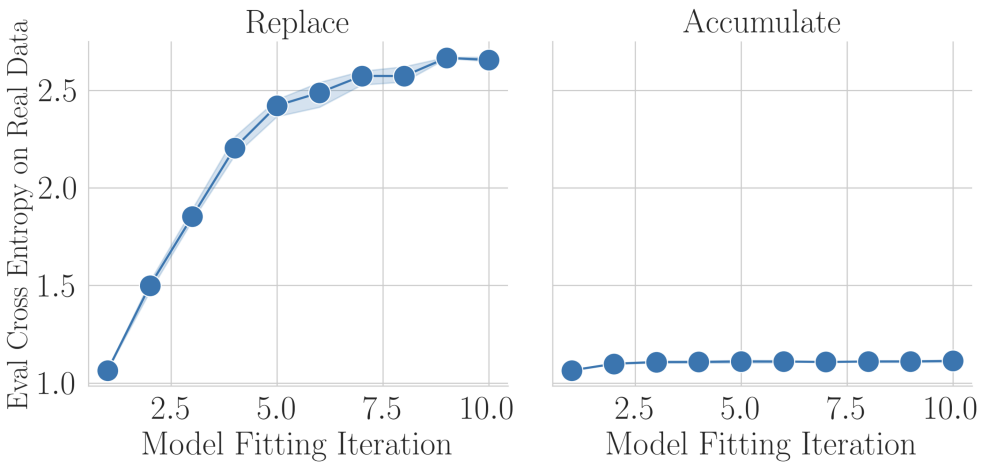

Figure 2: **Model Collapse in Supervised Finetuning of Language Models.** Finetuning Google's Gemma 2 2b [21] on Nvidia's HelpSteer 2 dataset [22] demonstrates that model collapse occurs if previous data are replaced after each model-fitting iteration (left), whereas model collapse is avoided if new synthetic data are instead accumulated with previous data (right).

1, right). More specifically, we find that if data are deleted the squared error between the fit mean $\hat{\mu}^{(n)}_{\text{Replace}}$ and the initial mean $\mu^{(0)}$ diverges (Fig. 1, middle left) and the fit covariance $\hat{\Sigma}^{(n)}_{\text{Replace}}$ relative to the initial covariance $\Sigma^{(0)}$ collapses to 0 (Fig. 1, bottom left), whereas if data accumulate, the squared error between the fit mean and the initial mean plateaus quickly (Fig. 1, middle right), as does the fit covariance relative to the initial covariance (Fig. 1, bottom right). Thus, deleting data causes model collapse, and accumulating data avoids model collapse.

Mathematically, in the univariate case, we are additionally able to characterize the limit distribution learned by the process described above:

Theorem 1. *For notational efficiency, for a univariate Gaussian, let $\hat{\mu}^{(t)}$ and $\hat{\sigma}^{(t)}$ denote $\hat{\mu}^{(t)}_{Accumulate}$ and $\hat{\Sigma}^{(t)}_{Accumulate}$. Suppose that the mean and covariance are updated as in Eqns. 3 and 4. Then*

$$\mathbb{E}\left(\sigma_t^2\right) = \sigma_0^2 \prod_{k=1}^{t}\left(1 - \frac{1}{nk^2}\right) \quad \xrightarrow{t\to\infty} \quad \sigma_0^2\left(\frac{\sin(\pi/\sqrt{n})}{\pi/\sqrt{n}}\right) \tag{5}$$

$$\mathbb{E}[(\mu_t - \mu_0)^2] = \sigma_0^2\left(1 - \prod_{k=1}^{t}\left(1 - \frac{1}{k^2 n}\right)\right) \quad \xrightarrow{t\to\infty} \quad \sigma_0^2\left(1 - \frac{\sin(\pi/\sqrt{n})}{\pi/\sqrt{n}}\right). \tag{6}$$

See App. Sec. A for the proof. This reveals two key differences when data accumulate: the covariance no longer collapses, and the mean no longer diverges, meaning model collapse is mitigated.

3 Model Collapse in Supervised Finetuning of Language Models

We next turn to the second setting for studying model collapse introduced by [20]: supervised finetuning of language models. We begin with an instruction following dataset – Nvidia's HelpSteer2 [22] – and repeatedly finetune a language model then sample new text data from it. For the language model, we use Google's Gemma 2 [21] because it is both high performing and relatively small. We again compare the two settings of interest: Replace and Accumulate. For Replace, we fine-tune the n-th language model only on data generated by the $(n-1)$-st language model. In Accumulate, we fine-tune the n-th language model on the original real data plus all the synthetic data sampled by all previously finetuned language models; thus, the amount of data that the nth model is finetuned on for Replace is constant $\sim 20k$, whereas the amount of data for Accumulate grows linearly $\sim 20k * n$.

We again find results consistent with multivariate Gaussian modeling and with Gerstgrasser et al. [10]: deleting data after each iteration leads to collapse, whereas accumulating data avoids collapse.

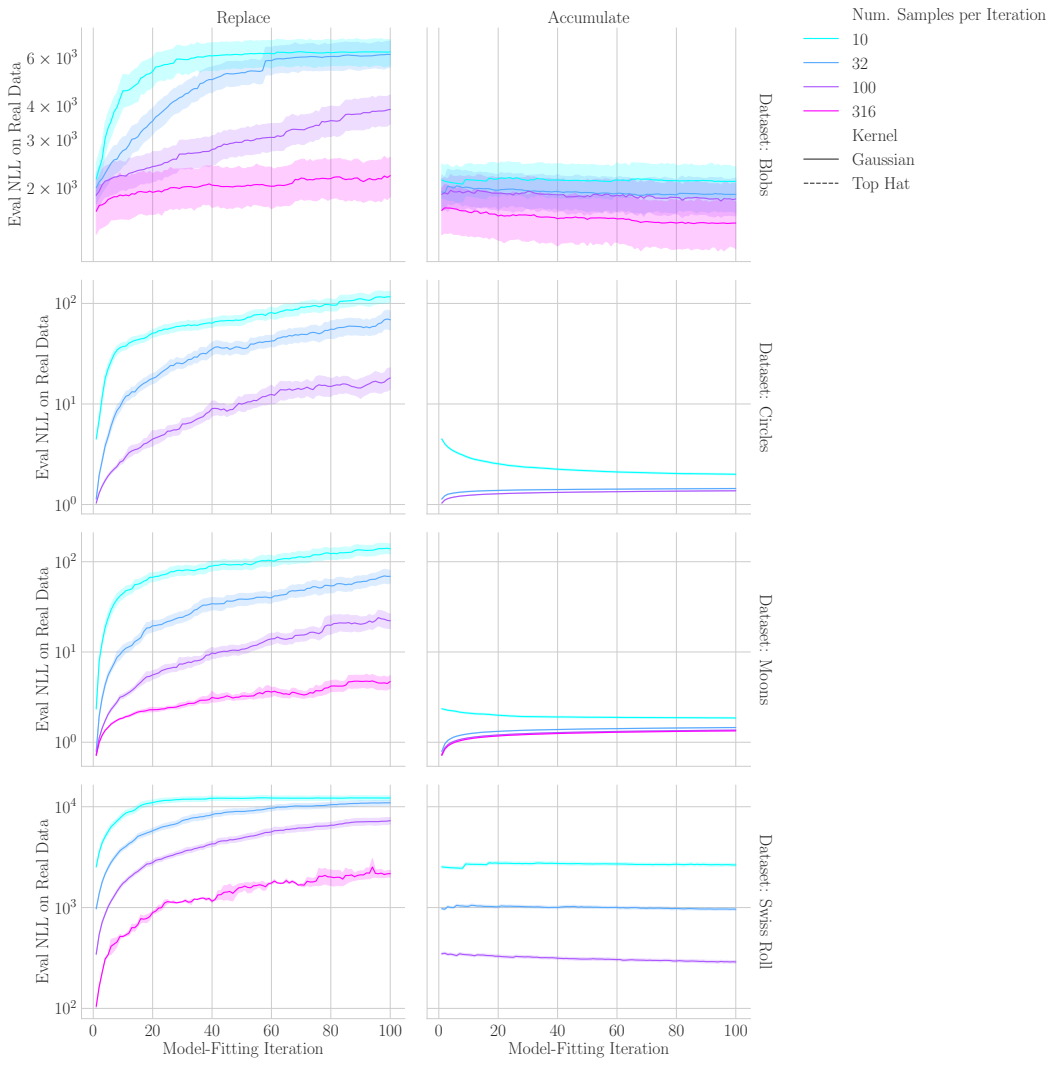

Figure 3: **Model Collapse in Kernel Density Estimation.** Deleting data en masse causes model collapse (left), whereas accumulating data avoid model collapse (right). **Interestingly, for specific combinations of datasets and number of samples per iteration, training on real and accumulating synthetic data yields lower loss on real test data than training on real data alone.**

4 Model Collapse in Kernel Density Estimation

We finally turn to the third setting for studying model collapsed introduced by [20]: kernel density estimation. Similar to multivariate Gaussian modeling, we begin with n real data points drawn from an initial probability distribution $p^{(0)}$: $X_1^{(0)}, ..., X_n^{(0)} \sim_{i.i.d.} p^{(0)}$. We then iteratively fit kernel density estimators to the data and sample new synthetic data from these estimators, again comparing Replace and Accumulate. In the Replace setting, we fit the kernel density estimator to n data samples from the most recently fit model, whereas in the Accumulate setting, we fit the estimator to all data points from all previous iterations, with the number of points growing linearly as $n(t+1)$:

$$\hat{p}^{(t+1)}_{\text{Replace}}(x) \stackrel{\text{def}}{=} \frac{1}{nh} \sum_{j=1}^{n} K\Big(\frac{x - X_j^{(t)}}{h}\Big) \tag{7}$$

$$\hat{p}^{(t+1)}_{\text{Accumulate}}(x) \stackrel{\text{def}}{=} \frac{1}{nh(t+1)} \sum_{i=0}^{t} \sum_{j=1}^{n} K\Big(\frac{x - X_j^{(i)}}{h}\Big) \tag{8}$$

where K is the kernel function and h is the bandwidth parameter. We consider two kernel functions: Gaussian and Top Hat. For sampling, at each iteration, we draw n new synthetic data points from the fitted kernel density estimators. We evaluate the performance using the negative log-likelihood (NLL) on real held-out test data; lower NLL indicates better performance. For data, we use four standard synthetic datasets from `sklearn` [7]: blobs, circles, moons, and swiss roll.

As in our previous experiments with multivariate Gaussian modeling and supervised finetuning of language models, we yet again observe the same result between replacing data and accumulating data (Fig. 3): replacing data causes a rapid increase in NLL as the number of model-fitting iterations increases, indicating that the kernel density estimators are becoming increasingly poor at modeling the true underlying distribution. This trend is consistent across both Gaussian and Top Hat kernels, and for different numbers of samples per iteration. In contrast, when data accumulate across model-fitting iterations, we observe that the NLL remains relatively stable, suggesting that accumulating data helps maintain the quality of the kernel density estimators.

Interestingly, for specific combinations of datasets and number of samples per iteration, *training on real plus accumulating synthetic data yields lower loss than training on real data alone* (Fig. 3, right column). Specifically, for Circles and Moons, sampling 10 synthetic data per model-fitting iteration and training on accumulating data yields lower test loss on real data, and for Swiss Roll, sampling 316 synthetic data per model-fitting iteration and training on accumulating data does so too. This is consistent with the language modeling results of Gerstgrasser et al. [10], but we know of no mechanism or theory to explain why performance can sometimes be improved with synthetic data. We leave that investigation to future work.

5 Discussion

Our findings support the claim that deleting data en masse after each iteration leads to model collapse, whereas accumulating data mitigates this issue. The consistency of these results across different model types and datasets suggests that *this distinction is a general phenomenon, and is not specific to any particular model or dataset or learning algorithm.*

The implication of these results is that under real-world dynamics, where new synthetic data is added to existing real and synthetic data, model collapse is unlikely. Our experiments are pessimistic, in the sense that real world practitioners filter data based on various indicators of data quality, e.g., [6, 23, 17, 13]; for a review, see Albalak et al. [1].

An especially interesting future direction is how to combine synthetic data generation with filtering techniques to enable performant and efficient pretraining at scale using synthetic data. As we saw in Sec. 4, training on accumulating real and synthetic data *can* improve performance on real test data. Identifying under what conditions, and why, this is possible is a tantalizing prospect.

References

[1] A. Albalak, Y. Elazar, S. M. Xie, S. Longpre, N. Lambert, X. Wang, N. Muennighoff, B. Hou, L. Pan, H. Jeong, C. Raffel, S. Chang, T. Hashimoto, and W. Y. Wang. A survey on data selection for language models, 2024. URL https://arxiv.org/abs/2402.16827.

[2] S. Alemohammad, J. Casco-Rodriguez, L. Luzi, A. I. Humayun, H. Babaei, D. LeJeune, A. Siahkoohi, and R. G. Baraniuk. Self-consuming generative models go mad. *arXiv preprint arXiv:2307.01850*, 2023.

[3] Q. Bertrand, A. J. Bose, A. Duplessis, M. Jiralerspong, and G. Gidel. On the stability of iterative retraining of generative models on their own data. *arXiv preprint arXiv:2310.00429*, 2023.

[4] M. Bohacek and H. Farid. Nepotistically trained generative-ai models collapse. *arXiv preprint arXiv:2311.12202*, 2023.

[5] M. Briesch, D. Sobania, and F. Rothlauf. Large language models suffer from their own output: An analysis of the self-consuming training loop. *arXiv preprint arXiv:2311.16822*, 2023.

[6] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei. Language models are few-shot learners, 2020. URL https://arxiv.org/abs/2005.14165.

[7] L. Buitinck, G. Louppe, M. Blondel, F. Pedregosa, A. Mueller, O. Grisel, V. Niculae, P. Prettenhofer, A. Gramfort, J. Grobler, R. Layton, J. VanderPlas, A. Joly, B. Holt, and G. Varoquaux. API design for machine learning software: experiences from the scikit-learn project. In *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, pages 108–122, 2013.

[8] E. Dohmatob, Y. Feng, and J. Kempe. Model collapse demystified: The case of regression. *arXiv preprint arXiv:2402.07712*, 2024.

[9] E. Dohmatob, Y. Feng, P. Yang, F. Charton, and J. Kempe. A tale of tails: Model collapse as a change of scaling laws. *arXiv preprint arXiv:2402.07043*, 2024.

[10] M. Gerstgrasser, R. Schaeffer, A. Dey, R. Rafailov, H. Sleight, J. Hughes, T. Korbak, R. Agrawal, D. Pai, A. Gromov, D. A. Roberts, D. Yang, D. L. Donoho, and S. Koyejo. Is model collapse inevitable? breaking the curse of recursion by accumulating real and synthetic data, 2024. URL https://arxiv.org/abs/2404.01413.

[11] Y. Guo, G. Shang, M. Vazirgiannis, and C. Clavel. The curious decline of linguistic diversity: Training language models on synthetic text. *arXiv preprint arXiv:2311.09807*, 2023.

[12] R. Hataya, H. Bao, and H. Arai. Will large-scale generative models corrupt future datasets? In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 20555–20565, 2023.

[13] J. Li, A. Fang, G. Smyrnis, M. Ivgi, M. Jordan, S. Gadre, H. Bansal, E. Guha, S. Keh, K. Arora, et al. Datacomp-lm: In search of the next generation of training sets for language models. *arXiv preprint arXiv:2406.11794*, 2024.

[14] M. Marchi, S. Soatto, P. Chaudhari, and P. Tabuada. Heat death of generative models in closed-loop learning, 2024. URL https://arxiv.org/abs/2404.02325.

[15] G. Martínez, L. Watson, P. Reviriego, J. A. Hernández, M. Juarez, and R. Sarkar. Combining generative artificial intelligence (ai) and the internet: Heading towards evolution or degradation? *arXiv preprint arXiv:2303.01255*, 2023.

[16] G. Martínez, L. Watson, P. Reviriego, J. A. Hernández, M. Juarez, and R. Sarkar. Towards understanding the interplay of generative artificial intelligence and the internet. *arXiv preprint arXiv:2306.06130*, 2023.

[17] G. Penedo, H. Kydlíček, L. B. allal, A. Lozhkov, M. Mitchell, C. Raffel, L. V. Werra, and T. Wolf. The fineweb datasets: Decanting the web for the finest text data at scale, 2024. URL https://arxiv.org/abs/2406.17557.

[18] M. E. A. Seddik, S.-W. Chen, S. Hayou, P. Youssef, and M. Debbah. How bad is training on synthetic data? a statistical analysis of language model collapse, 2024.

[19] I. Shumailov, Z. Shumaylov, Y. Zhao, Y. Gal, N. Papernot, and R. Anderson. The curse of recursion: Training on generated data makes models forget. *arXiv preprint arXiv:2305.17493*, 2023.

[20] I. Shumailov, Z. Shumaylov, Y. Zhao, N. Papernot, R. Anderson, and Y. Gal. Ai models collapse when trained on recursively generated data. *Nature*, 631(8022):755–759, 2024. ISSN 1476-4687. doi: 10.1038/s41586-024-07566-y. URL https://doi.org/10.1038/s41586-024-07566-y.

[21] G. Team, M. Riviere, S. Pathak, P. G. Sessa, C. Hardin, S. Bhupatiraju, L. Hussenot, T. Mesnard, B. Shahriari, A. Ramé, J. Ferret, P. Liu, P. Tafti, A. Friesen, M. Casbon, S. Ramos, R. Kumar, C. L. Lan, S. Jerome, A. Tsitsulin, N. Vieillard, P. Stanczyk, S. Girgin, N. Momchev, M. Hoffman, S. Thakoor, J.-B. Grill, B. Neyshabur, O. Bachem, A. Walton, A. Severyn, A. Parrish, A. Ahmad, A. Hutchison, A. Abdagic, A. Carl, A. Shen, A. Brock, A. Coenen, A. Laforge, A. Paterson, B. Bastian, B. Piot, B. Wu, B. Royal, C. Chen, C. Kumar, C. Perry, C. Welty, C. A. Choquette-Choo, D. Sinopalnikov, D. Weinberger, D. Vijaykumar, D. Rogozińska, D. Herbison, E. Bandy, E. Wang, E. Noland, E. Moreira, E. Senter, E. Eltyshev, F. Visin, G. Rasskin, G. Wei, G. Cameron, G. Martins, H. Hashemi, H. Klimczak-Plucińska, H. Batra, H. Dhand, I. Nardini, J. Mein, J. Zhou, J. Svensson, J. Stanway, J. Chan, J. P. Zhou, J. Carrasqueira, J. Iljazi, J. Becker, J. Fernandez, J. van Amersfoort, J. Gordon, J. Lipschultz, J. Newlan, J. yeong Ji, K. Mohamed, K. Badola, K. Black, K. Millican, K. McDonell, K. Nguyen, K. Sodhia, K. Greene, L. L. Sjoesund, L. Usui, L. Sifre, L. Heuermann, L. Lago, L. McNealus, L. B. Soares, L. Kilpatrick, L. Dixon, L. Martins, M. Reid, M. Singh, M. Iverson, M. Görner, M. Velloso, M. Wirth, M. Davidow, M. Miller, M. Rahtz, M. Watson, M. Risdal, M. Kazemi, M. Moynihan, M. Zhang, M. Kahng, M. Park, M. Rahman, M. Khatwani, N. Dao, N. Bardoliwalla, N. Devanathan, N. Dumai, N. Chauhan, O. Wahltinez, P. Botarda, P. Barnes, P. Barham, P. Michel, P. Jin, P. Georgiev, P. Culliton, P. Kuppala, R. Comanescu, R. Merhej, R. Jana, R. A. Rokni, R. Agarwal, R. Mullins, S. Saadat, S. M. Carthy, S. Perrin, S. M. R. Arnold, S. Krause, S. Dai, S. Garg, S. Sheth, S. Ronstrom, S. Chan, T. Jordan, T. Yu, T. Eccles, T. Hennigan, T. Kocisky, T. Doshi, V. Jain, V. Yadav, V. Meshram, V. Dharmadhikari, W. Barkley, W. Wei, W. Ye, W. Han, W. Kwon, X. Xu, Z. Shen, Z. Gong, Z. Wei, V. Cotruta, P. Kirk, A. Rao, M. Giang, L. Peran, T. Warkentin, E. Collins, J. Barral, Z. Ghahramani, R. Hadsell, D. Sculley, J. Banks, A. Dragan, S. Petrov, O. Vinyals, J. Dean, D. Hassabis, K. Kavukcuoglu, C. Farabet, E. Buchatskaya, S. Borgeaud, N. Fiedel, A. Joulin, K. Kenealy, R. Dadashi, and A. Andreev. Gemma 2: Improving open language models at a practical size, 2024. URL https://arxiv.org/abs/2408.00118.

[22] Z. Wang, Y. Dong, O. Delalleau, J. Zeng, G. Shen, D. Egert, J. J. Zhang, M. N. Sreedhar, and O. Kuchaiev. Helpsteer2: Open-source dataset for training top-performing reward models, 2024.

[23] A. Wettig, A. Gupta, S. Malik, and D. Chen. Qurating: Selecting high-quality data for training language models, 2024. URL https://arxiv.org/abs/2402.09739.

A Gaussian Model Fitting: Mathematical Results and Proofs

A.1 Setup

Lemma 2. *Using the notation of Theorem 1, we can express* $\mu_t = \sum_{r=1}^{t} \sigma_{r-1} \frac{\overline{z_r}}{r} + \mu_0$.

Proof. Note that $X_{i,t} = \mu_{t-1} + \sigma_{t-1} z_{i,t}$, where $z_{i,t} \sim \mathcal{N}(0,1)$. Therefore,

$$\begin{aligned}
\mu_t &= \frac{1}{nt} \sum_{r=1}^{t} \sum_{i=1}^{n} X_{i,r} \\
&= \frac{t-1}{t} \mu_{t-1} + \frac{\mu_{t-1}}{t} + \sigma_{t-1} \frac{\overline{z_t}}{t} \\
&= \mu_{t-1} + \sigma_{t-1} \frac{\overline{z_t}}{t}.
\end{aligned}$$

Therefore, $\mu_t = \sum_{r=1}^{t} \sigma_{r-1} \cdot \frac{\overline{z_r}}{r} + \mu_0$. □

Lemma 3. *Under the setup described in Theorem 1,* $\mathbb{E}[\frac{\sigma_t^2}{\sigma_0^2}] = \prod_{k=1}^{t} \left(1 - \frac{1}{nk^2}\right) \xrightarrow{t \to \infty} \frac{\sin(\pi/\sqrt{n})}{\pi/\sqrt{n}}$.

Proof. Using the recursive expression for μ_t in Lemma 2, we can rewrite

$$\begin{aligned}
\sigma_t^2 &= \frac{1}{nt} \sum_{r=1}^{t} \sum_{i=1}^{n} (X_{i,r} - \mu_t)^2 \\
&= \frac{1}{nt} \sum_{r=1}^{t} \sum_{i=1}^{n} \left(X_{i,r} - \overline{X_r} + \overline{X}_r - \mu_t\right)^2 \\
&= \frac{1}{nt} \sum_{r=1}^{t} \left(\sum_{i=1}^{n} \left(X_{i,r} - \overline{X_r}\right)^2 + n(\overline{X_r} - \mu_t)^2 \right) \\
&= \frac{1}{t} \sum_{r=1}^{t} \left(\sigma_{r-1}^2 S_r^2 + (\mu_{r-1} + \sigma_{r-1}\overline{z_r} - \mu_t)^2\right).
\end{aligned}$$

In the last line, we define $S_r^2 = \sum_{i=1}^{n} (X_{i,r} - \overline{X_r})^2$. The term

$$(\mu_{r-1} + \sigma_{r-1}\overline{z_r} - \mu_t)^2 = \left(\sigma_{r-1}\overline{z_r} - \sum_{k=r}^{t} \sigma_{k-1} \cdot \frac{\overline{z_k}}{k}\right)^2,$$

so

$$\begin{aligned}
\sigma_t^2 &= \frac{1}{t} \sum_{r=1}^{t} \left(\sigma_{r-1}^2 S_r^2 + \left(\sigma_{r-1}\overline{z_r} - \sum_{k=r}^{t} \sigma_{k-1} \frac{\overline{z_k}}{k}\right)^2 \right) \\
\Rightarrow t\sigma_t^2 &= \sum_{r=1}^{t} \left(\sigma_{r-1}^2 S_r^2 + \left(\sigma_{r-1}\overline{z_r}\left(1 - \frac{1}{r}\right) - \sum_{k=r+1}^{t} \sigma_{k-1} \frac{\overline{z_k}}{k}\right)^2 \right).
\end{aligned}$$

We now compute the conditional expectations of the terms in this sum. Where $\mathcal{F}_i$ denotes the ith filtration,

$$\mathbb{E}[\sigma_{r-1}^2 S_r^2 | \mathcal{F}_{t-1}] = \begin{cases} \sigma_{r-1}^2 S_r^2 & r < t \\ \sigma_{t-1}^2 \cdot \left(\frac{n-1}{n}\right) & r = t. \end{cases}$$

For $r = t$, we find that

$$\mathbb{E}\left[\left(\sigma_{r-1}\overline{z_r} \cdot \left(1 - \frac{1}{r}\right) - \sum_{k=r+1}^{t} \sigma_{k-1} \cdot \frac{\overline{z_k}}{k}\right)^2 | \mathcal{F}_{t-1}\right] = \sigma_{t-1}^2 \left(1 - \frac{1}{t}\right) \cdot \frac{1}{n}.$$

On the other hand, when $r < t$,

$$\mathbb{E}\left[\left(\sigma_{r-1}\overline{z_r}\cdot\left(1-\frac{1}{r}\right)-\sum_{k=r+1}^{t-1}\sigma_{k-1}\cdot\frac{\overline{z_k}}{k}-\sigma_{t-1}\cdot\frac{\overline{z}_t}{t}\right)^2|\mathcal{F}_{t-1}\right]$$
$$=\sigma_{t-1}^2\cdot\frac{1}{t^2}\cdot\frac{1}{n}+\left(\sigma_{r-1}\overline{z_r}\cdot\left(1-\frac{1}{r}\right)-\sum_{k=r+1}^{t-1}\sigma_{k-1}\cdot\frac{\overline{z_k}}{k}\right)^2.$$

Therefore,

$$\begin{aligned}\mathbb{E}[t\sigma_t^2|\mathcal{F}_{t-1}] &= (t-1)\sigma_{t-1}^2+\sigma_{t-1}^2\cdot\left(1-\frac{1}{n}\right)+\sigma_{t-1}^2\cdot\left(\frac{t-1}{t}\right)\cdot\left(\frac{1}{n}\right)+\sigma_{t-1}^2\cdot\left(1-\frac{1}{t}\right)^2\cdot\left(\frac{1}{n}\right)\\ &= \sigma_{t-1}^2\left(t-1+1-\frac{1}{n}+\frac{1}{tn}-\frac{1}{t^2n}+\frac{1}{n}-\frac{2}{tn}+\frac{1}{t^2n}\right)\\ &= \sigma_{t-1}^2\left(t-\frac{1}{tn}\right).\end{aligned}$$

It follows that

$$\mathbb{E}[\sigma_t^2|\mathcal{F}_{t-1}] = \sigma_{t-1}^2\left(1-\frac{1}{t^2n}\right) < \sigma_{t-1}^2$$

for all t. Thus, $\{\sigma_t^2\}_t$ is a supermartingale, and

$$\sigma_t^2 \xrightarrow{a.s.} \sigma_\infty^2$$

because σ_t^2 is bounded below by 0. Therefore, we still have convergence. Next, letting $m_t = \mathbb{E}[\sigma_t^2]$, we have

$$m_t = m_{t-1}\left(1-\frac{1}{t^2n}\right) = \cdots = \sigma_0^2\prod_{k=1}^{t}\left(1-\frac{1}{k^2n}\right),$$

so

$$\mathbb{E}[\sigma_t^2] = \sigma_0^2\prod_{k=1}^{\infty}\left(1-\frac{1}{k^2n}\right). \tag{9}$$

By a theorem of Euler, this is equal to

$$\sigma_0^2\frac{\sin(\pi/\sqrt{n})}{\pi/\sqrt{n}}. \tag{10}$$

□

Observe that by performing a variable replacement and using L'Hospital's rule, it is clear that $\lim_{n\to\infty}\mathbb{E}[\sigma_t^2] = \sigma_0^2$.

Finally, we are able to compute $\mathbb{E}[(\mu_t-\mu_0)^2]$.

Corollary 4. *The expected error in the mean*

$$\mathbb{E}[(\mu_t-\mu_0)^2] = \sigma_0^2\left(1-\prod_{k=1}^{t}\left(1-\frac{1}{k^2n}\right)\right). \tag{11}$$

Proof. Using the recursion from Lemma 2 and the expression for the variance in Lemma 4, we can rewrite

$$\begin{aligned}
\mathbb{E}[(\mu_t - \mu_0)^2] &= \sum_{k=1}^{t} \frac{\mathbb{E}[\sigma_{k-1}^2]}{nk^2} \\
&= \sigma_0^2 \sum_{k=1}^{t} \frac{1}{k^2 n} \prod \ell = 1^{k-1} \left(1 - \frac{1}{\ell^2 n}\right) \\
&= \sigma_0^2 \sum_{k=1}^{t} \left(\prod_{\ell=1}^{k-1} \left((1 - \frac{1}{\ell^2 n}\right) - \prod_{\ell=1}^{k} \left(1 - \frac{1}{\ell^2 n}\right) \right) \\
&= \sigma_0^2 \left(1 - \prod_{k=1}^{t} \left(1 - \frac{1}{k^2 n}\right)\right).
\end{aligned}$$

□

Therefore,

$$\lim_{t \to \infty} \mathbb{E}[(\mu_t - \mu_0)^2] = \sigma_0^2 \left(1 - \frac{\sin(\pi/\sqrt{n})}{\pi/\sqrt{n}}\right).$$