Is Model Collapse Inevitable? Breaking the Curse of Recursion by Accumulating Real and Synthetic Data

Matthias Gerstgrasser*[1,2], Rylan Schaeffer*[1,3], Apratim Dey*[1], Rafael Rafailov*[1], Henry Sleight[3], John Hughes[3], Tomasz Korbak[3], Rajashree Agrawal[3], Dhruv Pai[1], Andrey Gromov[4], Dan Roberts[5,6], Diyi Yang[1], David Donoho[1], and Sanmi Koyejo[1]

[1] Stanford University
[2] Harvard University
[3] Constellation
[4] University of Maryland, College Park
[5] Massachusetts Institute of Technology
[6] Sequoia Capital

Abstract. The proliferation of generative models, combined with pretraining on web-scale data, raises a timely question: what happens when models are trained on their own generated outputs? Recent investigations into model-data feedback loops proposed that such loops would lead to a phenomenon termed *model collapse*, under which performance progressively degrades with each model-data feedback iteration until fitted models become useless. However, those studies largely assumed that new data replace old data over time, where an arguably more realistic assumption is that data accumulate over time. In this paper, we ask: what effect does accumulating data have on model collapse? We empirically study this question by pretraining sequences of deep generative models (language models, diffusion models, variational autoencoders) on different tasks (causal language modeling, molecular conformation, image generation). After confirming that replacing the original real data by each generation's synthetic data does indeed tend towards model collapse, we discover that accumulating the successive generations of synthetic data alongside the original real data avoids model collapse. To understand why accumulating data can avoid model collapse, we use an analytically tractable framework of linear models introduced by prior work which showed replacing causes the test error to diverge; we extend their analysis to prove that if data instead accumulate, the test error has a finite upper bound independent of the number of iterations, meaning model collapse no longer occurs. Our work provides consistent empirical and theoretical evidence that not discarding real data avoids model collapse.

Keywords: Model Collapse · Model-Data Feedback Loops · Generative Models · Language Models · Diffusion Models · Variational Autoencoders

* Denotes equal authorship.

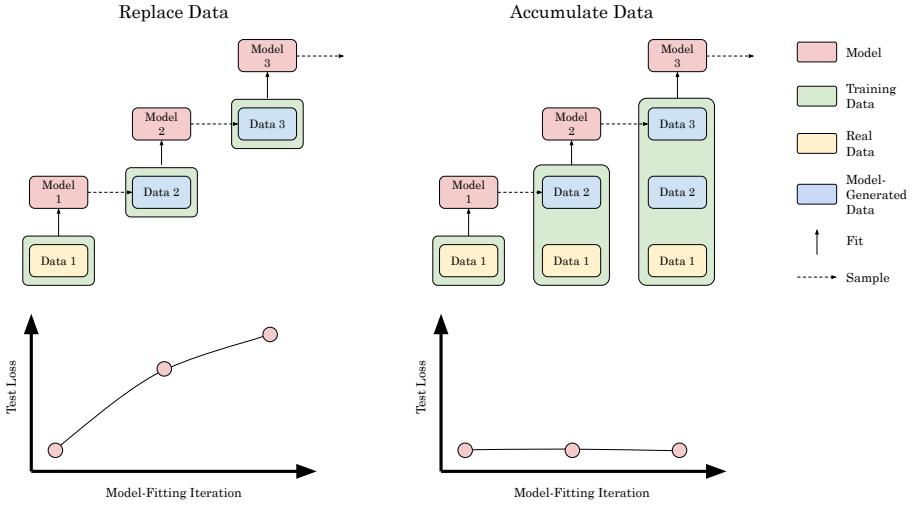

Fig. 1: Two Settings to Study Model Collapse. Model collapse is a phenomenon where sequences of generative models trained on their own outputs progressively degrade until the latest model becomes useless. Left: Many prior works studied model collapse where data are *replaced* with each model-fitting iteration. Right: We study model collapse where data *accumulate* with each iteration and demonstrate accumulating data avoids model collapse.

1 Introduction

The advent of large-scale generative models such as GPT-4 (1), DALL-E (22) and Stable Diffusion (24) has revolutionized the field of artificial intelligence. These models, trained on vast web-scale datasets, exhibit remarkable capabilities in generating text, images, and other media (6; 25). However, as these models become more widely used, an increasing amount of generated data populates the web. This raises a critical question: what are the consequences of training generative models on datasets containing their own outputs?

Recent studies have investigated this question, revealing that training generative models on their own outputs can cause performance to progressively degrade with each model-fitting iteration, eventually rendering newer models useless (2; 4; 5; 9; 10; 12; 18; 19; 27) (see App. A for review and discussion of prior work). This phenomenon was consequently labeled *model collapse*. Model collapse warns that democratizing access to generative models runs the risk of polluting the very data necessary to train future iterations of generative models. To better understand this phenomenon, most prior works have assumed each model's generated data *replaces* previous data. However, deleting real data en masse is uncommon, and models are often trained with increasing data over time – e.g., 1.4 trillion tokens for Llama 1 (30), 2 trillion for Llama 2 (31), 15 trillion for Llama 3 – in which presumably both human-generated and machine-generated data are accumulating in training sets scraped from the internet. It

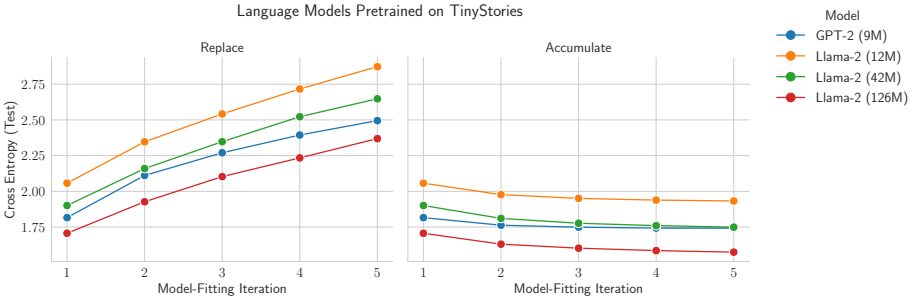

Fig. 2: Data Accumulation Avoids Model Collapse in Language Modeling. Sequences of causal transformer-based language models are pretrained on TinyStories (11). Cross-entropy validation loss increases when replacing data (left), but not when accumulating data (right). Synthetic data was sampled with temperature = 1.0.

was noted in some of those works (2; 4; 10; 12; 18) that model collapse can be either slowed down or negated by mixing real data in with generated data.

Motivated by this discrepancy, we study the effect of *accumulating* data on model collapse. Our data-accumulating setting is, in some sense, maximally pessimistic: it considers a hypothetical future where synthetic data are dumped on the internet to be scraped up for training future generative models. Nevertheless, we find that *model collapse is avoided when data accumulates*.

We begin by studying model collapse experimentally with deep generative models trained on realistic data: transformers on causal language modeling (Sec. 2.1), diffusion models on molecular conformation (Sec. 2.2) and variational autoencoders on images (Sec. 2.3). We confirm that replacing data at every iteration indeed causes test error to increase with the number of iterations but that accumulating synthetic data with real data avoids model collapse for all models and for all data modalities we test. We then turn to an analytically tractable framework of a sequence of linear models (9; 20) to demonstrate why replacing data and accumulating data have different effects, Altogether, our work suggests that data accumulation may be robust to model collapse and emphasizes the importance of considering accumulating data and other real-world data dynamics in the analysis of model collapse in generative models trained on web-scale data.

2 Accumulating Data Avoids Model Collapse in Deep Generative Models

2.1 Transformer-Based Causal Language Modeling

Experiments We first train causal transformers (32) on text data. Specifically, we pretrain 9M parameter GPT-2 (21) and 12M, 42M and 125M parameter Llama2 (31) language models for a single epoch on TinyStories (11), a 470M token GPT-3.5/4-generated dataset of short stories at a kindergarten reading

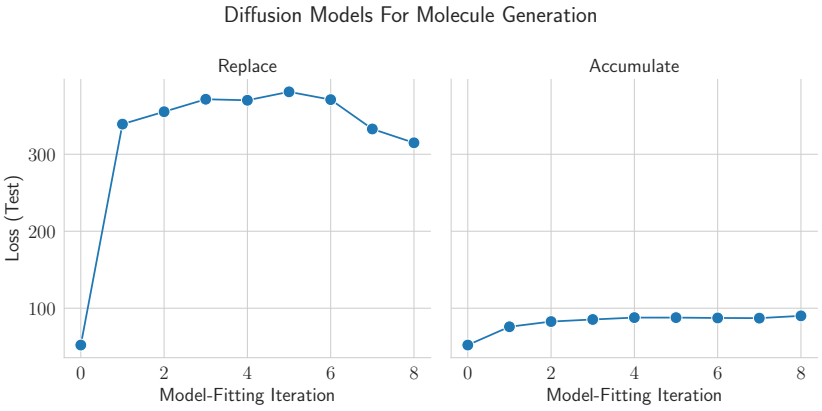

Fig. 3: Data Accumulation Avoids Model Collapse in Geometric Diffusion Modeling. GeoDiff, a diffusion-based molecular conformation generation model, is trained on a subset of Drugs data containing molecular structures found in drugs. Test loss degrades when replacing data (left) but not when accumulating data (right).

level. For each model-fitting iteration $n \geq 2$, we sample a new dataset of the same size as TinyStories from the previous language model and then either replace or concatenate the previous dataset with the newly generated dataset. In each model-fitting iteration, we then pretrain a newly initialized model on the replaced or concatenated dataset from the previous iteration. We experiment with sampling the new datasets using temperatures 0.3 or 1.0. We chose this combination of architectures, scales, dataset, and sampling as a good balance between being representative, being diverse and being computationally feasible.

Results For all architectures, parameter counts, and sampling temperatures, as the number of model-fitting iterations increased, replacing data led to an increase in test cross entropy (Fig. 2 top). In contrast, for all architectures, parameter counts, and sampling temperatures, as the number of model-fitting iterations increased, accumulating data led to equal-or-lower test cross entropy (Fig. 2 bottom). See Appendices D and G for ablations.

2.2 Diffusion Models on Molecular Conformation Data

Experiments We train sequences of diffusion models on molecular conformation data. Specifically, we train GeoDiff (35), a geometric diffusion model for molecular conformation generation, on the GEOM-Drugs (3) dataset. We down-sample the training split of GEOM-Drugs to 40,000 molecular conformations, which we use as our initial training set, and perform 50 diffusion steps for each prediction. For the loss, we use the standard loss used by GeoDiff: a weighted variational lower bound to the conditional likelihood; for more details, see (35).

Results Over 8 model-fitting iterations, we find test loss increases when replacing data, matching our language model experiments, and test loss remains relatively

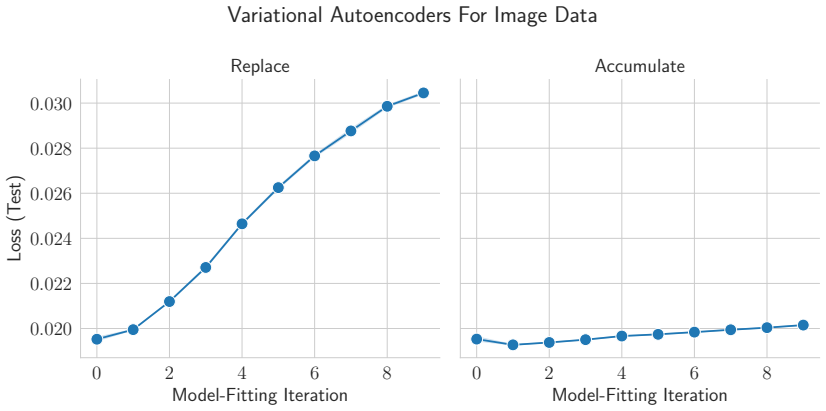

Fig. 4: Data Accumulation Avoids Model Collapse in Variational Autoencoders for Image Generation. Sequences of variational autoencoders (VAEs) are trained on CelebA, a large-scale dataset of human faces. Test loss degrades when replacing data (left) but not when accumulating data (right).

Fig. 5: Sampled Images from Left: Replacing data with data generated by the previous iteration's newly trained VAE yields lower quality and eventually leads to complete mode collapse. Middle: Accumulating data with data generated by the previous iteration's newly trained VAE preserves the quality and diversity of generated data across iterations. Right: Baseline samples after 100 training epochs on the dataset.

constant when accumulating data (Fig. 3). Unlike with language models, we found that when replacing data, performance worsens significantly in the first model-fitting iteration trained on synthetic data and does not degrade further.

2.3 Variational Autoencoders on Image Data

Experiments We lastly train sequences of variational autoencoders (VAEs) (14; 23) on CelebA (16), chosen as a balance between being a realistic dataset with many samples, color images and resolution, and computational feasibility of training multiple iterations of models on accumulating data. The loss is the

standard VAE loss: reconstruction error plus the KL divergence between the encoder's Gaussian and the isotropic Gaussian prior. See Appendix E for details.

Results We find that replacing data at each iteration again exhibits model collapse: the test error rises with each additional iteration, and each iteration yields lower quality and less diverse generated faces until all model generations represent a single mode (Figure 5 left). In contrast, accumulating data at each iteration significantly slows model collapse: the test error increases significantly slower with each additional iteration. We discuss further in Appendix E.

3 Accumulating Data Avoids Model Collapse in Linear Models

To gain mathematical understanding, we employ an analytical framework introduced in prior work (9; 20) which considers a sequence of linear models that are fit to synthetic data sampled from previously fit linear models. See App. B for details of the framework, our results and corresponding proofs. We will show that the framework produces the same types of test error behaviors measured empirically for these two data-use strategies. Within this framework, Dohmatob et al. (9) proved that if data are replaced, the test error is given by:

$$E_{\text{test}}^{\text{Replace}}(\hat{w}_n) = \frac{\sigma^2 d}{T - d - 1} \times n \tag{1}$$

where σ^2 is the noise per dimension, d is the number of features and T is the number of samples per iteration. Critically, when data are replaced, the test error grows linearly with the number of iterations n. In contrast, when data accumulate, we prove the test error is upper bounded *regardless of the number of iterations* n:

$$E_{\text{test}}^{\text{Accum}}(\hat{w}_n) \leq \frac{\sigma^2 d}{T - d - 1} \times \frac{\pi^2}{6} \tag{2}$$

To explain intuitively, when previous data are discarded, the model is more strongly affected by the new noise that each iteration of generated data introduces, and adds that to the effects experienced in earlier iterations. But when data are accumulated, because iteration i contributes fraction $1/i$ to the training dataset, the additional noise from the ith iteration is decreased overall. This suggests that accumulating generated data with real data can indeed avoid model collapse. We numerically simulate the the analytical results and find an almost perfect match (App. Fig. 15).

4 Discussion

This work explored the phenomenon of model collapse, an important concern as AI-generated content permeates the internet and future training datasets.

Bibliography

[1] Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F.L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al.: Gpt-4 technical report. arXiv preprint arXiv:2303.08774 (2023) 2

[2] Alemohammad, S., Casco-Rodriguez, J., Luzi, L., Humayun, A.I., Babaei, H., LeJeune, D., Siahkoohi, A., Baraniuk, R.G.: Self-consuming generative models go mad. arXiv preprint arXiv:2307.01850 (2023) 2, 3, 10, 11, 12, 13

[3] Axelrod, S., Gomez-Bombarelli, R.: Geom, energy-annotated molecular conformations for property prediction and molecular generation. Scientific Data **9**(1), 185 (2022) 4

[4] Bertrand, Q., Bose, A.J., Duplessis, A., Jiralerspong, M., Gidel, G.: On the stability of iterative retraining of generative models on their own data. arXiv preprint arXiv:2310.00429 (2023) 2, 3, 10, 12

[5] Briesch, M., Sobania, D., Rothlauf, F.: Large language models suffer from their own output: An analysis of the self-consuming training loop. arXiv preprint arXiv:2311.16822 (2023) 2, 10

[6] Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al.: Language models are few-shot learners. Advances in neural information processing systems **33**, 1877–1901 (2020) 2

[7] Caponnetto, A., De Vito, E.: Optimal rates for the regularized least-squares algorithm. Foundations of Computational Mathematics **7**, 331–368 (2007) 15

[8] Cui, H., Loureiro, B., Krzakala, F., Zdeborová, L.: Generalization error rates in kernel regression: The crossover from the noiseless to noisy regime. Advances in Neural Information Processing Systems **34**, 10131–10143 (2021) 15

[9] Dohmatob, E., Feng, Y., Kempe, J.: Model collapse demystified: The case of regression. arXiv preprint arXiv:2402.07712 (2024) 2, 3, 6, 10, 13, 14, 15, 16, 17, 29

[10] Dohmatob, E., Feng, Y., Yang, P., Charton, F., Kempe, J.: A tale of tails: Model collapse as a change of scaling laws. arXiv preprint arXiv:2402.07043 (2024) 2, 3, 10, 11, 12

[11] Eldan, R., Li, Y.: Tinystories: How small can language models be and still speak coherent english? arXiv preprint arXiv:2305.07759 (2023) 3

[12] Hataya, R., Bao, H., Arai, H.: Will large-scale generative models corrupt future datasets? In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 20555–20565 (2023) 2, 3, 10

[13] Jain, A., Montanari, A., Sasoglu, E.: Scaling laws for learning with real and surrogate data. arXiv preprint arXiv:2402.04376 (2024) 11

[14] Kingma, D.P., Welling, M.: Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114 (2013) 5

[15] Kwon, W., Li, Z., Zhuang, S., Sheng, Y., Zheng, L., Yu, C.H., Gonzalez, J., Zhang, H., Stoica, I.: Efficient memory management for large language model serving with pagedattention. In: Proceedings of the 29th Symposium on Operating Systems Principles. pp. 611–626 (2023) 21
[16] Liu, Z., Luo, P., Wang, X., Tang, X.: Deep learning face attributes in the wild. In: Proceedings of International Conference on Computer Vision (ICCV) (December 2015) 5
[17] Marchi, M., Soatto, S., Chaudhari, P., Tabuada, P.: Heat death of generative models in closed-loop learning (2024) 11
[18] Martínez, G., Watson, L., Reviriego, P., Hernández, J.A., Juarez, M., Sarkar, R.: Combining generative artificial intelligence (ai) and the internet: Heading towards evolution or degradation? arXiv preprint arXiv:2303.01255 (2023) 2, 3, 10, 11, 12
[19] Martínez, G., Watson, L., Reviriego, P., Hernández, J.A., Juarez, M., Sarkar, R.: Towards understanding the interplay of generative artificial intelligence and the internet. arXiv preprint arXiv:2306.06130 (2023) 2, 10, 12
[20] Mobahi, H., Farajtabar, M., Bartlett, P.: Self-distillation amplifies regularization in hilbert space. Advances in Neural Information Processing Systems **33**, 3351–3361 (2020) 3, 6, 10, 13, 29
[21] Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., et al.: Language models are unsupervised multitask learners. OpenAI blog **1**(8), 9 (2019) 3
[22] Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., Chen, M.: Hierarchical text-conditional image generation with clip latents. arXiv preprint arXiv:2204.06125 **1**(2), 3 (2022) 2
[23] Rezende, D.J., Mohamed, S., Wierstra, D.: Stochastic backpropagation and approximate inference in deep generative models. In: International conference on machine learning. pp. 1278–1286. PMLR (2014) 5
[24] Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10684–10695 (2022) 2
[25] Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E.L., Ghasemipour, K., Gontijo Lopes, R., Karagol Ayan, B., Salimans, T., et al.: Photorealistic text-to-image diffusion models with deep language understanding. Advances in neural information processing systems **35**, 36479–36494 (2022) 2
[26] Seddik, M.E.A., Chen, S.W., Hayou, S., Youssef, P., Debbah, M.: How bad is training on synthetic data? a statistical analysis of language model collapse (2024) 11
[27] Shumailov, I., Shumaylov, Z., Zhao, Y., Gal, Y., Papernot, N., Anderson, R.: The curse of recursion: Training on generated data makes models forget. arXiv preprint arXiv:2305.17493 (2023) 2, 10
[28] Simon, J.B., Dickens, M., Deweese, M.: Neural tangent kernel eigenvalues accurately predict generalization (2021) 15

[29] Taori, R., Hashimoto, T.: Data feedback loops: Model-driven amplification of dataset biases. In: International Conference on Machine Learning. pp. 33883–33920. PMLR (2023) 10

[30] Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., et al.: Llama: Open and efficient foundation language models. arXiv preprint arXiv:2302.13971 (2023) 2

[31] Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., et al.: Llama 2: Open foundation and fine-tuned chat models. arXiv preprint arXiv:2307.09288 (2023) 2, 3

[32] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017) 3

[33] Wei, A., Hu, W., Steinhardt, J.: More than a toy: Random matrix models predict how real-world neural representations generalize. In: International Conference on Machine Learning. pp. 23549–23588. PMLR (2022) 15

[34] Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., et al.: Huggingface's transformers: State-of-the-art natural language processing. arXiv preprint arXiv:1910.03771 (2019) 21

[35] Xu, M., Yu, L., Song, Y., Shi, C., Ermon, S., Tang, J.: Geodiff: A geometric diffusion model for molecular conformation generation. arXiv preprint arXiv:2203.02923 (2022) 4

A Summarization and Discussion of Prior and Related Work

Prior Empirical Work A growing body of recent work has investigated the phenomenon of iteratively training models on data generated by previous models, e.g., (2; 4; 5; 9; 10; 12; 18; 19; 27) and (in a different context) (29). (12) and (19) conducted experiments replacing real training data with generated data at each iteration, assuming that the dataset size remains fixed over time. They found that this iterative retraining procedure can lead to model degradation if the proportion of synthetic data becomes too high. Similarly, (27) ran experiments with Gaussian mixture models, VAEs, and language models in which the total number of samples per iteration was held constant, and the samples always originated with the previous model rather than aggregating over time. Building on this work, (2) considered three treatments of data: fully replacing real data with synthetic data, augmenting a fixed real dataset with additional synthetic data, and mixing new real data with synthetic data at each iteration. In almost all of their experiments, they drew a fixed size dataset from the most recent model at each iteration, without accumulating data. (4) also assumed that dataset size and mixing proportions are constant over time in their theoretical stability analysis and empirical validation.

Prior Theoretical Work Over the last few years, there has been significant research effort contributing to our theoretical understanding of model behavior when synthetic data are integrated into training. The most closely related works to ours are Dohmatob et al. (9) and (10); of course, the inspiration for the linear regression model studied in this paper directly comes from Dohmatob et al. (9). Dohmatob et al. (9) performs an in-depth analysis of high dimensional linear and ridge regression when the training data used per iteration are generated from the previous iteration's fitted model. They are able to conclude that the test error grows linearly with the iteration count in their setup, as well as derive more interesting and more nuanced results using random matrix theory. They also discuss how to mitigate model collapse through optimal regularization both when the training data are noise-free and noisy versions of the previous model's synthetic outputs. A related noise-free setup was studied by (20) in the case of self-distillation. Although (20) considers a more general setup with ridge regression as a special case, they use *noiseless* predictions from the previous model as the training data for the next model, and show that eventually, the predictions shrink to zero. Through this, they highlight that self-distillation induces regularization in the function space, which initially is beneficial for reducing over-fitting, but eventually over-regularization causes underfitting and hence performance decay. (10) go beyond the linear model to study model collapse – they study the tails of LLM outputs vs. real data and provide scaling laws that clearly identify regimes of model degradation when synthetic data misses *tails* present in real data. They identify an interesting *phase transition* in the test error scaling law depending on the size of the real dataset size in comparison to (a functional of) the chopped-off tail, and conclude that enough real data is able to mitigate model

collapse. All these works consider the scenario where the amount of training data available per iteration is fixed (and does not grow with the iteration count), and it is certainly possible that with larger amount of synthetic data (from prediction by the previous model), several of these scalings would improve significantly. For example, in Equation (12) of (10), one obtains the linear scaling (with iteration count) of test error simply because the amount of synthetic data generated per iteration is the same. If one generated synthetic data with size proportional to the iteration count, then at iteration n, the scaling would, instead of n, be like $n^{1-c}/(1-c)$ for $c < 1$. When one does not increase the dataset size, (10) points out that increasing the proportion of real data would help one to avoid model collapse altogether. However, even if one *did* increase the amount of synthetic data with iteration count, Theorem 3.2 coupled with Corollary 3.3 in (10) would tell us that the *amount* of real data was all that mattered – if the amount of real data is large, we overcome model collapse. If one only had synthetic data (and no real data), no matter how large, it would be impossible to regain the original real-data scaling laws. The scenario we study is highly inspired by these pioneering works, but still, in our view, different. We consider the case when we keep *augmenting* synthetic data (generated by the previous model trained on all the previous data so far) as iterations progress, much akin to how – in our view – the internet evolves. We observe that we can avoid model collapse in this setting. The analysis of previous models in our case is more involved, since the data used for training at iteration n is not homogeneous – different models from the past impart different statistical aspects to different parts of the training data. We also note a related augmentation model studied by (13) – they perform risk minimization augmenting real data with synthetic data available from a potentially different independent source. One of their messages is that augmentation of (even) pure noise can be looked upon as adding a ridge penalty and hence, in certain cases, can improve test error. Their setup, however, is different from ours, since the synthetic data in their setup is not obtained by a learning algorithm employed on the real data, and the process is not iterative. However, morally, each iteration of ours involves risk minimization on data statistically composed of an equal mixture of data generated from the previous models, and hence each iteration of ours can be mapped to the general framework developed in (13), although the dependencies among the various models trained in our setup introduce theoretical complications that do not seem to be too easily addressed by the theory developed in (13). Shortly after v1 of our manuscript was uploaded to ArXiv, two other manuscripts appeared, dealing with the theoretical aspects in a setting similar to ours. Theorem 1 of (17) obtains the same square summability scaling of the variance as us. (26) studies collapse in language models in both purely synthetic and partly synthetic regimes and obtains deviation bounds as model iterations progress.

Considering Accumulating Data The two papers we found that partially considered accumulating data are (18) and (2). (2) did so in one-half of one experiment: StyleGAN2 trained on FliqrFaces 128×128 (App. Fig. 6). The authors concluded that accumulating data does not avoid model collapse, but merely slows it down.

However, we believe that a closer examination of their results (App. Fig. 6) reveals that accumulating data causes the test error to plateau to a relatively low error with increasing numbers of model-fitting iterations. This result would support our conclusion that accumulating data avoids model collapse and does not merely delay it. The results from (18) are harder to evaluate; model collapse only seems to occur when the amount of synthetic data added per model-fitting iteration is 2× the total amount of accumulated data, and the subsequent work by the authors switched from accumulating data to replacing data (19). We think understanding what conditions and why these discrepancies exist is an interesting future direction.

Avoiding Model Collapse Several papers present methods for avoiding or slowing model collapse. (4) shows in the replacing data setting that model collapse will not occur if the initial generative models approximate the data distribution well enough and the proportion of real data is sufficiently large with respect to the synthetic data. (10) similarly demonstrates that in the replacing data setting, carefully selecting real data to mix with synthetic data can avoid model collapse. Other solutions may also be possible in various models and under various assumptions. To our knowledge, no paper has claimed an "optimal" strategy to avoid model collapse, and neither has ours.

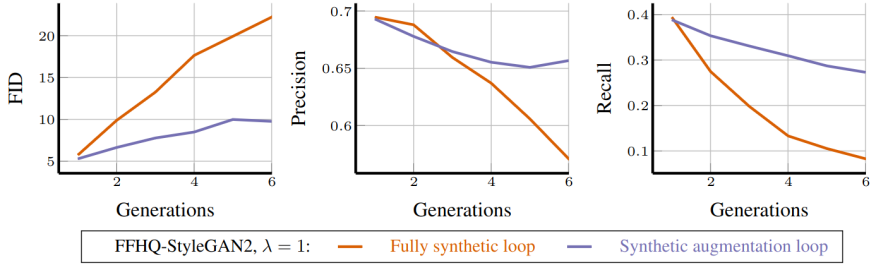

Fig. 6: Clarification of Data Accumulation in (2). Figure 7 from (2) (above) shows that linearly accumulating data ("Synthetic augmentation loop") causes poor behavior to plateau with the number of model-fitting iterations. (2) write, "Our experiments [...] support our main conclusion [that] fixed real training data only delays the inevitable degradation of the quality or diversity of the generative models over generations." We believe is that our evidence and their evidence is more consistent with the conclusion that accumulating data *avoids* model collapse and does not merely delay it.

B Accumulating Data Avoids Model Collapse in Linear Models

To gain mathematical understanding and intuition, we employ an analytical framework introduced in prior work (9; 20) to understand the difference between data accumulation and data replacement. We will show that it predicts the same types of test error behaviors for these two data-use strategies that were measured empirically. The framework considers a sequence of linear models that are fit to the synthetic data sampled from the linear generative model model based on the previously fit linear models. Within this framework, Dohmatob et al. (9) showed that if data are replaced across model-fitting iterations, then the test squared error increases linearly[7] with the number of iterations n. Here, we extend Dohmatob et al. (9)'s argument to show that if data instead accumulate across model-fitting iterations, then the test squared error is upper bounded by a relatively small constant, meaning model collapse is avoided[8].

The content in this section relies heavily on the framework and pioneering contributions of Dohmatob et al. (9). Our contribution is to study a *different way to use synthetic data in training*, namely accumulate, which seems to better align with certain real-world considerations. We show that our empirical results could have been anticipated on theoretical grounds, by applying the same analysis framework as in Dohmatob et al. (9), but instead to this specific training dataset pattern. We use the same framework to analyze some other ways that synthetic data might have be used, such as replace, again the theory aligns with many empirical results.

[7] To echo an earlier footnote, an approach 'halfway' between the 'replace' and 'accumulate' approaches would replace the previous dataset with a pure synthetic dataset of size iT at the i-th iteration. Analyzing this goes mostly in parallel, except the $1/i^2$ mentioned in running text now becomes $1/i$ for the 'halfway' approach. Consequently, the MSE scaling becomes $MSE \asymp O(\log(n))$; the 'halfway' approach with pure synthetic data but more of it, again has test error growing unboundedly with iterations. Thanks to Elvis Dohmatob, Yunzhen Feng and Julia Kempe for communicating this observation. See Appendix F for an extended discussion.

[8] In this theoretical section, we identify the term *model collapse* with the situation where test error diverges to infinity (at any rate) as iterations progress. Other authors may employ similar terminology while identifying it with different properties of test error. For example, (2) use the term MAD to refer to the situation where the distance between the distribution of the original data and that of the subsequent generative models grow farther apart, without necessarily diverging.

B.1 Notation and Preliminaries

Original Data Distribution. We adapt notations from (9). Define the distribution P_{Σ,w,σ^2} on $\mathbb{R}^d \times \mathbb{R}$ given by $(x, y) \sim P_{\Sigma,w,\sigma^2}$ iff :

$$\begin{aligned} &\text{(Input)} \quad x \sim \mathcal{N}(0, \Sigma), \\ &\text{(Noise)} \quad \epsilon \sim \mathcal{N}(0, \sigma^2), \text{ independent of } x, \\ &\text{(Label)} \quad y = x \cdot w^* + \epsilon. \end{aligned}$$

The positive integer d is the input-dimension, the matrix $\Sigma \in \mathbb{R}^{d\times d}$ is the true covariance structure of the input x, the vector w^* is the true linear relationship used to generate the original data and the scalar σ is the level of label noise. We start at iteration $n = 1$ with T initial independent data points (x_i, y_i) each following P_{Σ,w^*,σ^2}, that is, $y_i = x_i \cdot w^* + \epsilon_i$ for each $i = 1, 2, \cdots, T$. We form the design matrix $X \in^{T\times d}$ with $x_1^\top, \cdots, x_T^\top$ as rows. We also form the vectors Y and E with i-th coordinate y_i and ϵ_i respectively. In whatever follows, we will assume that X has full column rank, i.e., $T \geq d$, $X^\top X$ is invertible and the model is underparameterized.

Synthetic Data Generation Process. We generate synthetic data from the following sequence of distributions

$$P_{\Sigma,w^*,\sigma^2} \to P_{\Sigma,\hat{w}_1,\sigma^2} \to \ldots \to P_{\Sigma,\hat{w}_n,\sigma^2},$$

where $n \in \mathbb{N}$ is the number of iterations. The scheme is outlined as follows.

- For $n = 1$:
  - Accumulating Covariates/Features: $\tilde{X}_1 X$
  - Accumulating Targets: $\tilde{Y}_1 \hat{Y}_1 X w^* + E_1$, where $E_1 E \sim \mathcal{N}(0, \sigma^2 I_T)$
  - Fit linear model: $\hat{w}_1 = \tilde{X}_1^\dagger \tilde{Y}_1$
  - Sample synthetic data for the next iteration: $\hat{Y}_2 X \hat{w}_1 + E_2$, where $E_2 \sim \mathcal{N}(0, \sigma^2 I_T)$
- For $n \geq 2$:
  - Accumulating Covariates/Features: $\tilde{X}_n^\top = [\tilde{X}_{n-1}^\top; X^\top] \in \mathbb{R}^{d\times nT}$
  - Accumulating Targets: $\tilde{Y}_n^\top = [\tilde{Y}_{n-1}^\top; \hat{Y}_n^\top] \in \mathbb{R}^{1\times nT}$
  - Fit linear model: $\hat{w}_n \tilde{X}_n^\dagger \tilde{Y}_n$
  - Sample synthetic data for the next iteration: $\hat{Y}_{n+1} X \hat{w}_n + E_{n+1}$, where $E_{n+1} \sim \mathcal{N}(0, \sigma^2 I_T)$

Here, for a matrix A with full column rank, $A^\dagger = (A^\top A)^{-1} A^\top$ is the Moore-Penrose pseudo-inverse of A. The noise terms $E_1, E_2, \ldots, E_n$ are independent of each other and of the covariates/features. Since X has full column rank, so does $\tilde{X}_n$ for every $n \geq 1$.

Test Error. We are interested in the dynamics of the test error $E_{\text{test}}(\hat{w}_n)$ of this sequence of linear model $\hat{w}_1, \hat{w}_2, \ldots$. Note that evaluation of the model is done on the true distribution P_{Σ,w^*,σ^2}, even though the model is trained on the accumulated synthetic data. For any linear estimator $\hat{w}$ computed from the training data, we measure test error in the standard way:

$$E_{test}(w)\mathbb{E}\left[(x_{test}^T w - y_{test})^2\right] - \sigma^2 = [\|w - w^*\|_\Sigma^2] \tag{3}$$

where the expectation is taken over the training data and $(x_{test}, y_{test}) \sim P_{\Sigma,w^*,\sigma^2}$ independent of the training data.

A Note on Extensions to Ridge Regression and Kernel Methods. To reiterate a comment made previously by Dohmatob et al. (9), although we present our results in the context of ordinary linear regression in $\mathbb{R}^d$, our analysis can be readily extended to ridge regression and the kernel setting (7; 8; 28; 33). We focus here on a simple useful model for studying model collapse.

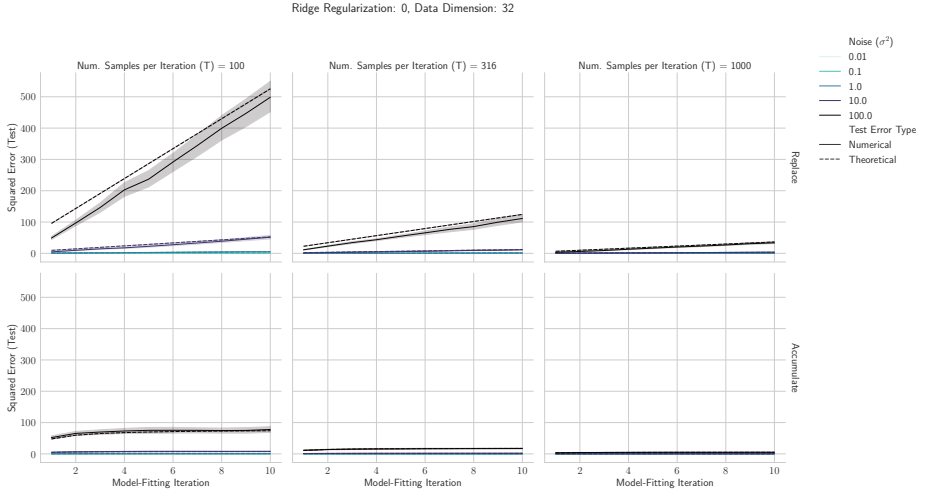

Fig. 7: Accumulating Data Avoids Model Collapse in Linear Regression. We consider sequences of linear models recurrently fit to generated targets by previous iterations of models. Top: If each linear model is fit to the generated targets of *only* the preceding linear model, i.e., data are replaced, then the test error grows linearly with the number of iterations n. Bottom: If each linear model is instead fit to the generate targets of *all* the preceding linear models, i.e., data accumulate, then the test error has a finite upper bound independent of the number of iterations. This suggests that data accumulation might be a robust solution for mitigating model collapse. For log test error and higher iterations, see Appendix Fig. 15.

B.2 Precise Test Error Characterization Under Accumulating Data

Our goal is to establish an analytic formula for the test error of the nth model in the data accumulation setting. We begin by characterizing the relationship between the fitted linear parameters $\hat{w}_n$ and the true parameters w^*. We remind the reader that we assume that X has full column rank, i.e., $X^\top X$ is invertible. Proofs are deferred to App. C.

Theorem 1. *In the data accumulation setting, $\forall n \geq 1$, the fitted linear parameters $\hat{w}_n$ can be expressed as:*

$$\hat{w}_n = w^* + (X^\top X)^{-1} X^\top \left(\sum_{i=1}^{n} \frac{E_i}{i} \right) \tag{4}$$

where, recall, w^* is the true parameter, X is the original design matrix, and E_i is the extra noise added at the i'th iteration.

Theorem 2. *For an n-fold synthetic data generation process with $T \geq d+2$ samples per iteration and isotropic features (ΣI_d), the test error for the ridgeless linear predictor $\hat{w}_n$ learned on the accumulated data up to iteration n is given by:*

$$E_{test}^{Accum}(\hat{w}_n) = \frac{\sigma^2 d}{T-d-1} \left(\sum_{i=1}^{n} \frac{1}{i^2} \right) \leq \frac{\sigma^2 d}{T-d-1} \times \frac{\pi^2}{6} \tag{5}$$

where, recall, σ^2 is the noise variance of the fake data generation process, d is the input dimension, and T is the number of samples (i.e., data points) added per iteration.

How does test error with accumulating data compare against test error with replacing data? Under otherwise identical assumptions, Dohmatob et al. (9) proved in the data-replacing setting that the test error is given by[9]:

$$E_{\text{test}}^{\text{Replace}}(\hat{w}_n) = \frac{\sigma^2 d}{T-d-1} \times n \tag{6}$$

When data are replaced, the test error grows linearly with the number of iterations n (Fig 7 top), with the rate of growth determined by a noise-to-signal ratio: the amount of noise per dimension σ^2 times the number of dimensions d, adjusted by the (per-iteration) sample size T. In contrast, when data accumulate, Theorem 2 shows the test error is upper bounded *regardless of the number of iterations n*:

$$E_{\text{test}}^{\text{Accum}}(\hat{w}_n) \leq \frac{\sigma^2 d}{T-d-1} \times \frac{\pi^2}{6}$$

This striking difference can be intuitively explained by the differences in the way data are handled across iterations. In the data replacement setting, because

[9] For notational simplicity, we assume that Dohmatob et al. (9)'s $T_0 T$ and $\sigma_0 \sigma$.

previous data were discarded, the model is more strongly affected by the new noise that each iteration of generated data introduces, and adds that to the effects experienced in earlier iterations. But in the data accumulation setting, because iteration i contributes fraction $1/i$ to the training dataset, the additional noise from the ith iteration of synthetic data has its effect on the model MSE shrunken proportional to $1/i^2$ (due to squared error). The summability of $1/i^2$ prevents the test error from growing indefinitely. This suggests that accumulating generated data with real data can indeed avoid model collapse.

C Proofs of Mathematical Results

We point out a lemma useful to prove Theorem 2.

Lemma 1. *Let T and d be positive integers with $T \geq d+2$, and let $X \in \mathbb{R}^{T\times d}$ be a random matrix with i.i.d. rows from $\mathcal{N}(0,\Sigma)$ with Σ positive definite. Then, X has full rank a.s. Moreover, it holds that:*

$$\mathbb{E}_X[(X^\top X)^{-1}] = \frac{1}{T-d-1}\Sigma^{-1}. \tag{7}$$

Proof. See Dohmatob et al. (9).

Assuming Lemma 1 and Theorem 1, we present the proof of Theorem 2.

Proof (Proof of Theorem 2). From Theorem 1, we have:

$$\hat{w}_n = w^* + (X^\top X)^{-1}X^\top \left(\sum_{i=1}^{n} \frac{E_i}{i}\right) \tag{8}$$

where w^* is the true parameter, X is the original data matrix, and E_i are the noise terms at each iteration, with $E_i \sim \mathcal{N}(0,\sigma^2 I_T)$. The test error is given by:

$$E_{\text{test}}(\hat{w}_n) = \mathbb{E}[||\hat{w}_n - w^*||_\Sigma^2] \tag{9}$$

where the expectation is taken over all random quantities involved.

Substituting $\hat{w}_n$ into the test error expression and using the fact that ΣI_d, we get:

$$\begin{aligned} E_{\text{test}}(\hat{w}_n) &= \mathbb{E}\left[\left(\sum_{i=1}^{n}\frac{E_i}{i}\right)^\top X(X^\top X)^{-2}X^\top\left(\sum_{i=1}^{n}\frac{E_i}{i}\right)\right] \\ &= \mathbb{E}\left[\sum_{i=1}^{n}\frac{\sigma^2}{i^2}\text{tr}(X(X^\top X)^{-2}X^\top)\right] \\ &= \sum_{i=1}^{n}\frac{\sigma^2}{i^2}\mathbb{E}\left[\text{tr}((X^\top X)^{-1})\right] \end{aligned}$$

Using Lemma 1, we have:

$$\mathbb{E}_X\left[\text{tr}((X^\top X)^{-1})\right] = \frac{d}{T-d-1} \tag{10}$$

Therefore, the test error for ridgeless regression with isotropic features in the data accumulation setting is:

$$E_{\text{test}}(\hat{w}_n) = \sum_{i=1}^{n} \frac{\sigma^2}{i^2} \cdot \frac{d}{T-d-1} < \frac{\sigma^2 d}{T-d-1}\left(\frac{\pi^2}{6}\right)$$

as $\sum_{i=1}^{n} i^{-2} < \sum_{i=1}^{\infty} i^{-2} = \pi^2/6$.

Finally, we prove Theorem 1.

Proof (Proof of Theorem 1).

We prove this theorem by induction.

Base case: For $n = 1$, we have:

$$\hat{w}_1 = \tilde{X}_1^\dagger \tilde{Y}_1 = (X^\top X)^{-1} X^\top (Xw^* + E_1) = w^* + (X^\top X)^{-1} X^\top E_1$$

which satisfies the lemma.

Inductive step: Assume that for some $n \geq 1$, we have:

$$\hat{w}_n = w^* + (X^\top X)^{-1} X^\top \left(\sum_{i=1}^{n} \frac{E_i}{i}\right)$$

Now, consider $\hat{w}_{n+1}$:

$$\begin{aligned}
\hat{w}_{n+1} &= \tilde{X}_{n+1}^\dagger \tilde{Y}_{n+1} \\
&= (\tilde{X}_{n+1}^\top \tilde{X}_{n+1})^{-1} \tilde{X}_{n+1}^\top \tilde{Y}_{n+1} \\
&= \frac{1}{n+1} (X^\top X)^{-1} \sum_{i=1}^{n+1} X^\top \hat{Y}_i
\end{aligned}$$

Recalling that $\hat{Y}_i$:

$$\hat{Y}_i = \begin{cases} Xw^* + E_1, & i = 1 \\ X\hat{w}_{i-1} + E_i, & 2 \leq i \leq n+1 \end{cases}$$

Substituting this back into the expression for $\hat{w}_{n+1}$:

$$\begin{aligned}\hat{w}_{n+1} &= \frac{1}{n+1}(X^\top X)^{-1}\left(X^\top(Xw^* + E_1) + \sum_{i=2}^{n+1} X^\top(X\hat{w}_{i-1} + E_i)\right)\\ &= \frac{1}{n+1}(X^\top X)^{-1}\left(X^\top X w^* + X^\top E_1 + \sum_{i=2}^{n+1}(X^\top X\hat{w}_{i-1} + X^\top E_i)\right)\\ &= \frac{1}{n+1}(X^\top X)^{-1}\left(X^\top X w^* + X^\top E_1 + \sum_{i=1}^{n}(X^\top X\hat{w}_{i} + X^\top E_{i+1})\right)\\ &= \frac{1}{n+1}(X^\top X)^{-1}\left(X^\top X w^* + \sum_{i=1}^{n} X^\top X\hat{w}_i + \sum_{i=1}^{n+1} X^\top E_i\right)\end{aligned}$$

Now, using the induction hypothesis:

$$\begin{aligned}\hat{w}_{n+1} &= \frac{1}{n+1}(X^\top X)^{-1}\left(X^\top X w^* + \sum_{i=1}^{n} X^\top X\left(w^* + (X^\top X)^{-1}X^\top\sum_{j=1}^{i}\frac{E_j}{j}\right) + \sum_{i=1}^{n+1} X^\top\right.\\ &= \frac{1}{n+1}(X^\top X)^{-1}\left((n+1)X^\top X w^* + \sum_{i=1}^{n} X^\top X(X^\top X)^{-1}X^\top\sum_{j=1}^{i}\frac{E_j}{j} + \sum_{i=1}^{n+1} X^\top E_\right.\\ &= w^* + \frac{1}{n+1}(X^\top X)^{-1}\left(\sum_{i=1}^{n} X^\top\sum_{j=1}^{i}\frac{E_j}{j} + \sum_{i=1}^{n+1} X^\top E_i\right)\\ &= w^* + \frac{1}{n+1}(X^\top X)^{-1}X^\top\left(\sum_{i=1}^{n}\sum_{j=1}^{i}\frac{E_j}{j} + \sum_{i=1}^{n+1} E_i\right)\end{aligned}$$

Now, we need to simplify the term $\sum_{i=1}^{n}\sum_{j=1}^{i}\frac{E_j}{j} + \sum_{i=1}^{n+1} E_i$. We can do this by counting the number of times each E_i appears in the double sum: E_1 appears n times in the double sum and once in the single sum, so its coefficient is $\frac{n+1}{1}$. E_2 appears $n-1$ times in the double sum and once in the single sum, so its coefficient is $\frac{n}{2}$. This continues along till we reach E_n, which appears once in the double sum and once in the single sum, so its coefficient is $\frac{2}{n}$. E_{n+1} appears only once in the single sum, so its coefficient is $\frac{1}{n+1}$. Therefore,

$$\sum_{i=1}^{n}\sum_{j=1}^{i}\frac{E_j}{j} + \sum_{i=1}^{n+1} E_i = \sum_{i=1}^{n+1}\frac{n+2-i}{i}E_i = (n+1)\sum_{i=1}^{n+1}\frac{E_i}{i}$$

Substituting this back into the expression for $\hat{w}_{n+1}$:

$$\begin{aligned}\hat{w}_{n+1} &= w^* + \frac{1}{n+1}(X^\top X)^{-1}X^\top\left((n+1)\sum_{i=1}^{n+1}\frac{E_i}{i}\right)\\ &= w^* + (X^\top X)^{-1}X^\top\sum_{i=1}^{n+1}\frac{E_i}{i}\end{aligned}$$

Therefore, by mathematical induction, the lemma holds for all $n \geq 1$.

D Additional Details and Ablations on Language Model Experiments

Implementation Details

Model training was implemented using Huggingface Transformers (34). Dataset generation was implemented using vllm (15).

Additional Plots

In addition to Figure **??** in the main text, Figures 8-11 show learning curves in larger print, with x-axes showing either epochs or gradient steps, and with axes shown in linear-linear or log-log scale, respectively.

Ablations

In addition to the experiments shown in the main paper, we conducted several ablation studies.

Controlling for dataset size. One possible concern is that when accumulating data, the train dataset size will grow at each model-fitting iteration, meaning subsequent models will be trained on more aggregate data than their counterparts in the replacement regime. To control for this, we run experiments controlling for this. In this "replace-multiple" regime, we create a fully synthetic dataset at the end of each model-fitting iteration, but grow the size of this dataset to match that of the accumulated data in the accumulation regime. Table 1 rightmost column shows that in this regime, evaluation loss still increases over model-fitting iterations.

Generation temperature. Most of our language model experiments were run with sampling temperature 1.0 during generation of new datasets. To ensure that this choice is not critical, we also run one experiment with temperature 0.3, and see that this shows similar results (with even larger increases in validation loss in the replacement regime than temperature 1.0), as shown in Table 1, row 2, and Figure 12.

Dataset size and training epochs. We similarly vary the size of the initial (and subsequent) training datasets and number of training epochs, and see that this has no qualitative effect on the results (Table 1, rows 3 & 4 show training on 1/5th of the TinyStories dataset for 1 & 3 epochs, respectively).

Model quality after first model-fitting iteration. Finally, we control specifically for model (and thus synthetic dataset) quality after the first iteration, to rule out an undue influence of a "bad" first synthetic dataset on subsequent training. Figure 13 shows performance in subsequent iterations for different amounts of training in the first iteration, showing no qualitative differences.

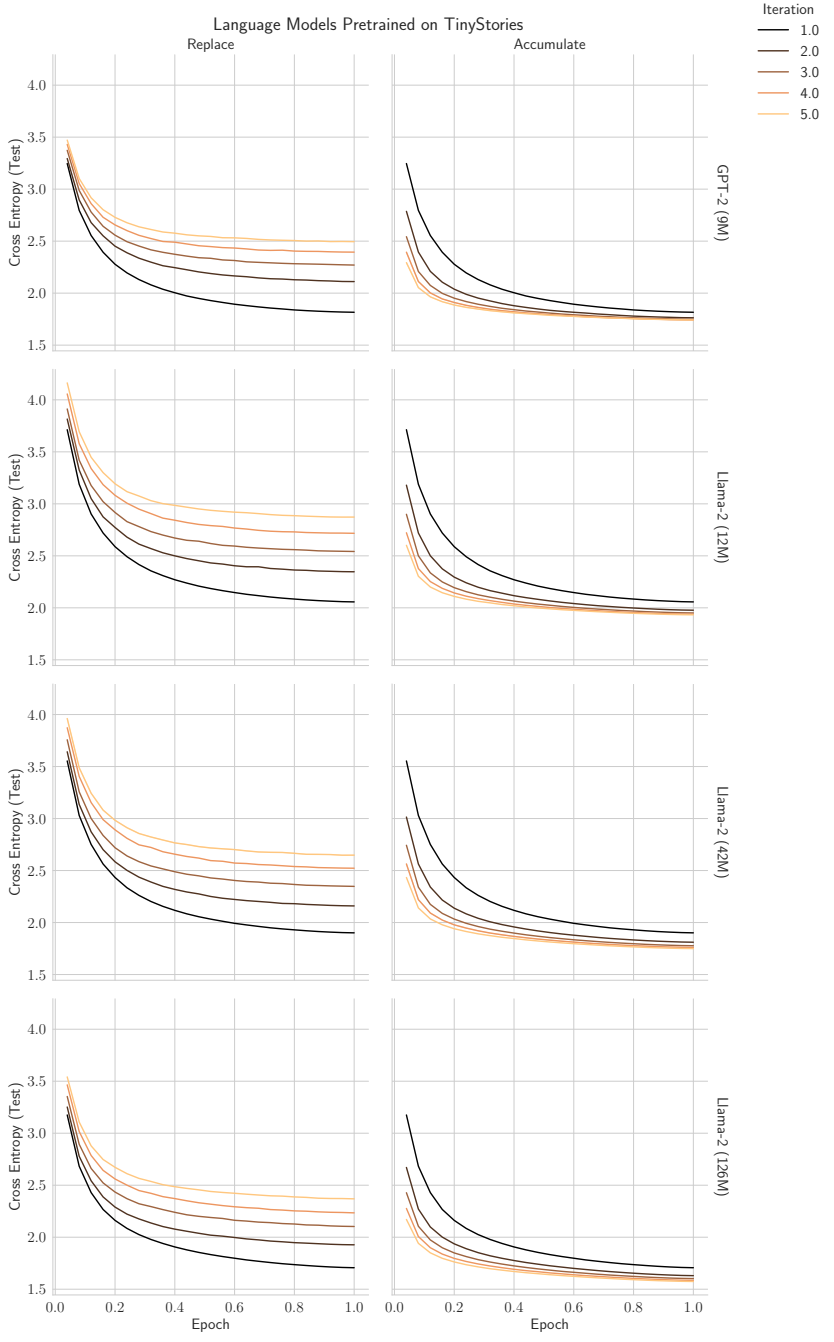

Fig. 8: Data Accumulation Avoids Model Collapse in Language Modeling. Learning curves for individual model-fitting iterations when repeatedly *replacing* data (left), and when *accumulating* data (right). Note: Epochs correspond to more gradient steps for accumulate than replace because the number of training data grows for accumulate.

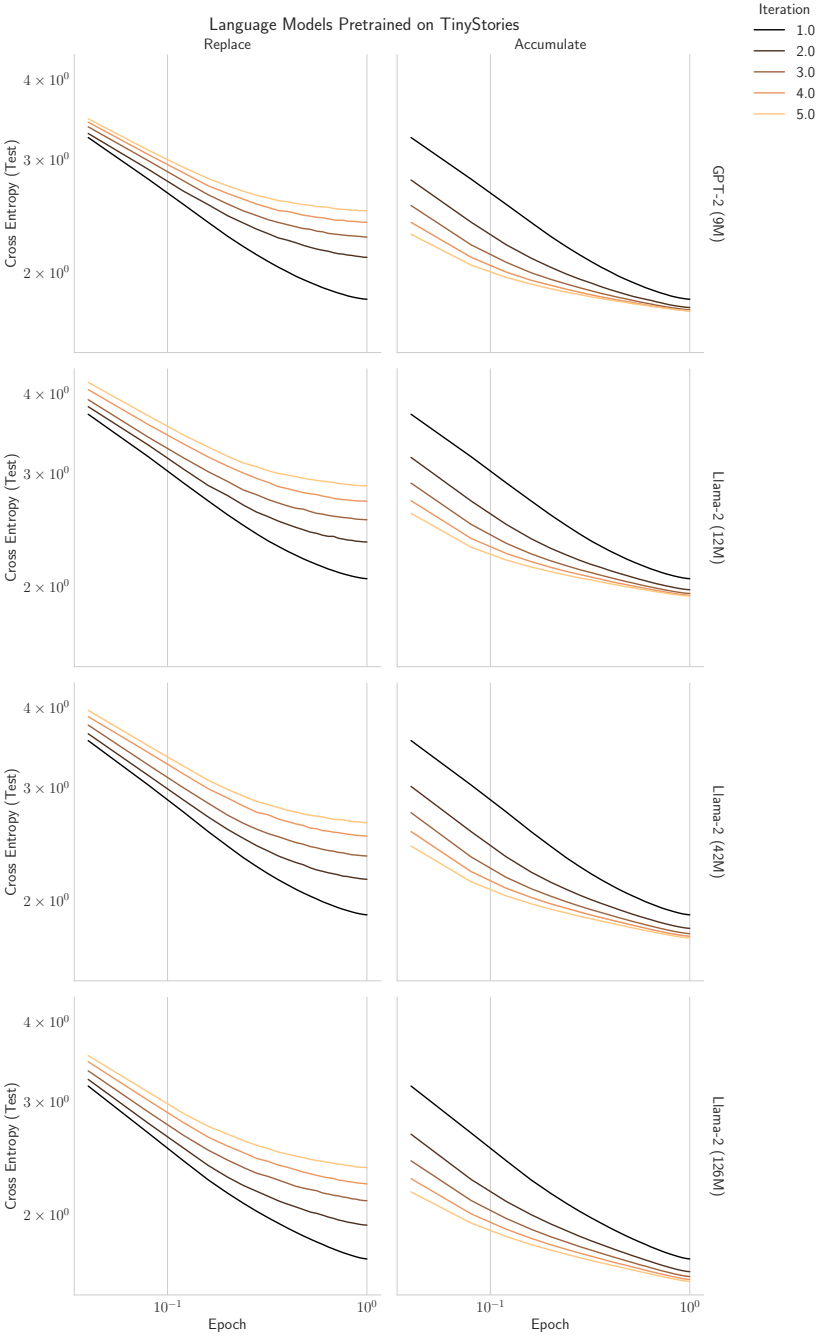

Fig. 9: Data Accumulation Avoids Model Collapse in Language Modeling. Learning curves for individual model-fitting iterations when repeatedly *replacing* data (left), and when *accumulating* data (right), in log-log scale. Note: Epochs correspond to more gradient steps for accumulate than replace because the number of training data grows for accumulate.

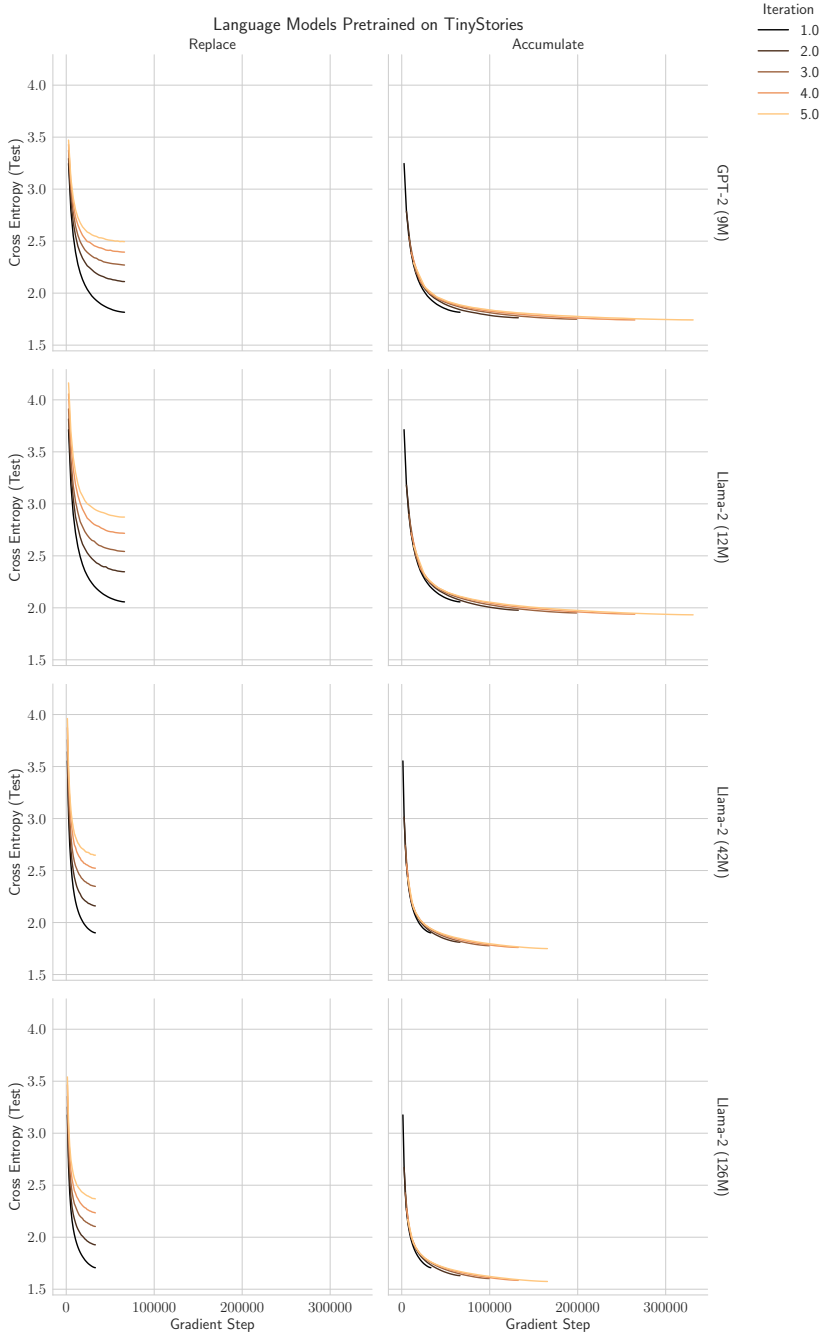

Fig. 10: Data Accumulation Avoids Model Collapse in Language Modeling. Learning curves for individual model-fitting iterations when repeatedly *replacing* data (left), and when *accumulating* data (right).

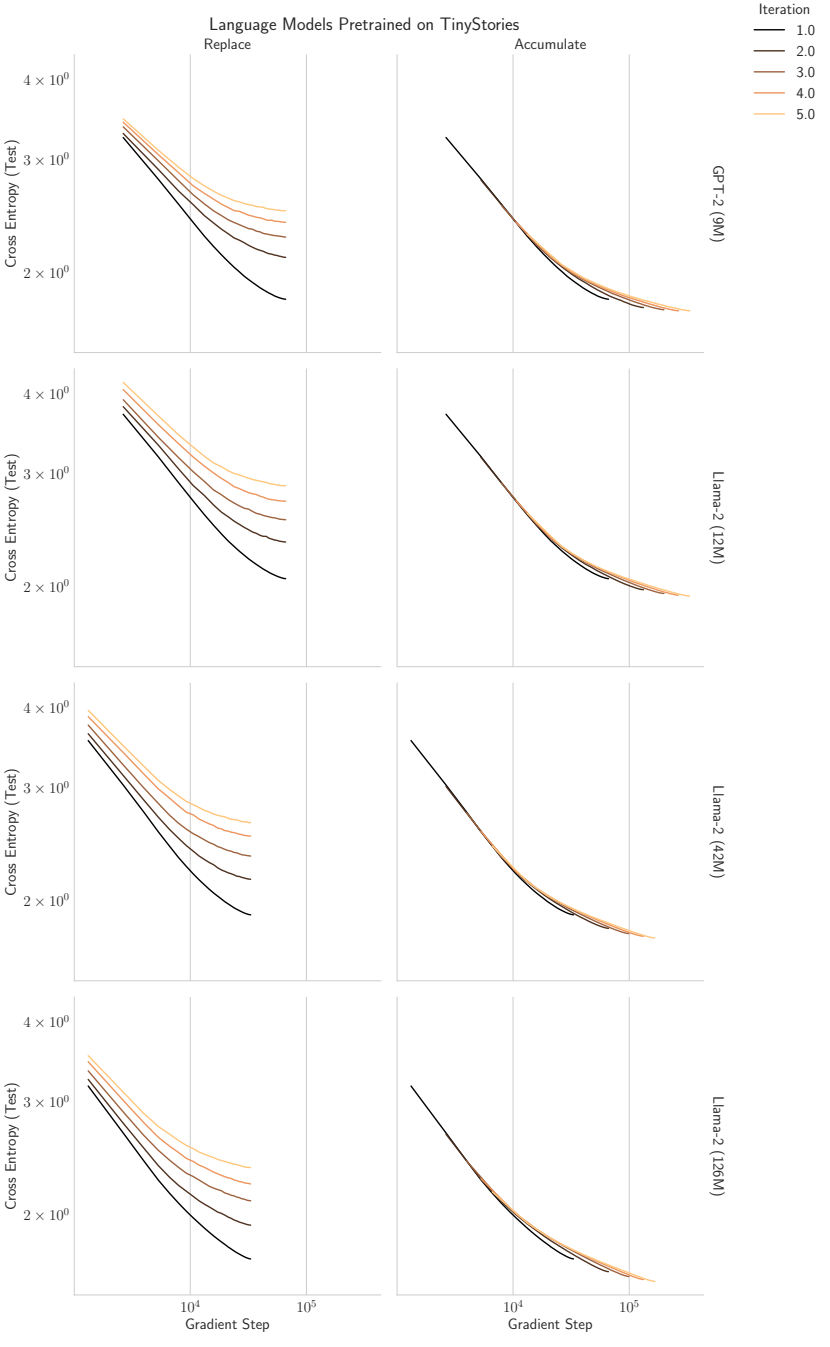

Fig. 11: Data Accumulation Avoids Model Collapse in Language Modeling. Learning curves for individual model-fitting iterations when repeatedly *replacing* data (left), and when *accumulating* data (right), in log-log scale.

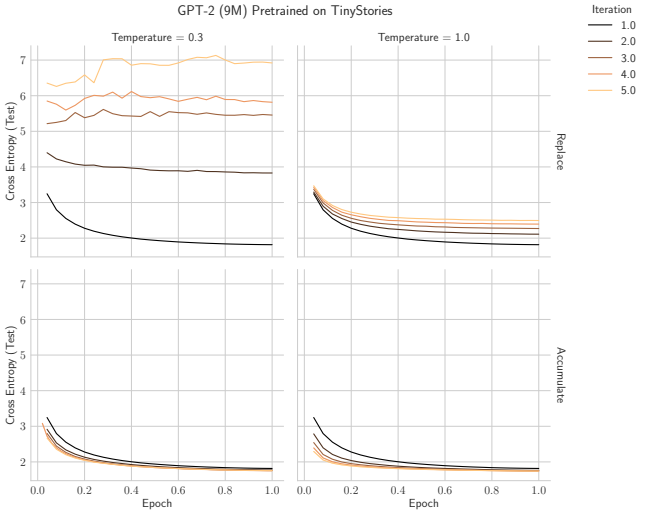

Fig. 12: Accumulating data shows stable behavior across different generation temperatures for a GPT-2 (9M) model, while replacing data does not.

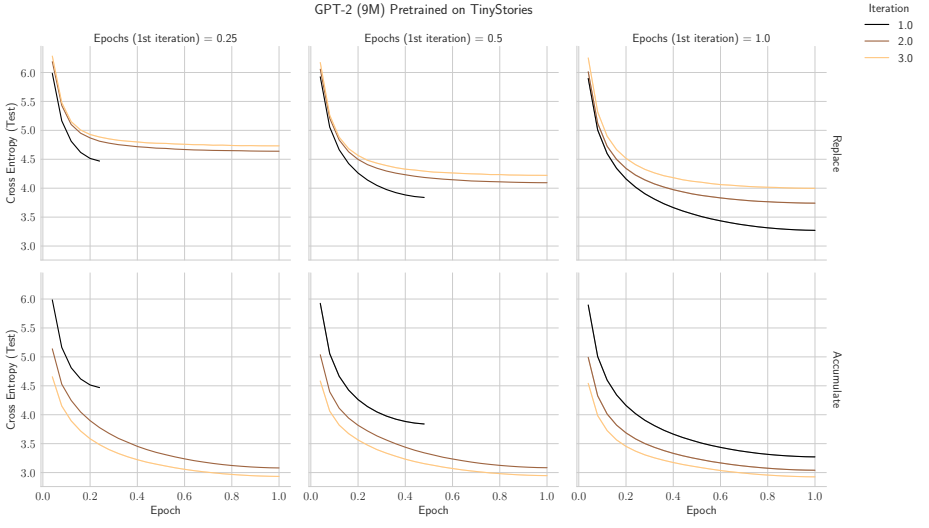

Fig. 13: Model quality after the first model-fitting iteration does not qualitatively change behavior in subsequent iterations. Columns show differing training amount (as measure by epochs) in first iteration.

Model	t=1	t=4 (acc)	t=4 (repl)	t=10 (repl)	t=4 (*)
GPT-2 (9M)	1.82	1.74 (-0.07)	2.39 (+0.58)	2.91 (+1.09)	2.18 (+0.36)
GPT-2 (9M) (temp=0.3)	1.82	1.75 (-0.06)	5.82 (+4.00)	9.85 (+8.04)	n/a
GPT-2 (9M) (small dataset)	2.56	2.28 (-0.28)	3.21 (+0.65)	3.72 (+1.16)	2.91 (+0.35)
ibid (+ 3 epochs)	1.99	1.87 (-0.12)	2.62 (+0.63)	n/a	n/a
Llama-2 (12M)	2.06	1.94 (-0.12)	2.72 (+0.66)	n/a	n/a
Llama-2 (42M)	1.90	1.76 (-0.14)	2.52 (+0.62)	n/a	n/a
Llama-2 (126M)	1.71	1.59 (-0.12)	2.23 (+0.53)	n/a	n/a

Table 1: Evaluation cross-entropy loss for different models at model-fitting iterations 1, 4 and 10 for replacement and accumulation regimes. (*) indicates a replacement regime with growing dataset size to ablate for total train set size.

E Additional Details on VAE Experiments

Experiment Details. As pre-processing, we crop and down-sample the images to 64x64 pixels. We use a standard convolutional architecture for the VAE model consisting of 5 convolutional layers with 32, 64, 128, 256, and 512 channels, respectively, and a similar convolutional decoder structure. The latent space is 128-dimensional isotropic Gaussian, represented by 2 MLP layers. Each data iteration consists of 100 training epochs, after which we generate 163K new training images by sampling latents from the Gaussian prior and the passing them through the generator model.

Analysis of Reconstructions. Figure 14 shows reconstructions after replacing (left) and accumulating (center) data, compared to baseline (right). Analyzing the reconstruction of test set images also reveals interesting findings - the model trained only on data from the prior iteration has indeed collapsed and cannot represent any other classes besides the single mode it generates. Interestingly, the model trained on aggregated data still maintains it's capabilities and generates accurate reconstructions, including smaller details such as glasses and hats. We hypothesize that this model maintains it's generative capabilities, but these details become a more minor sub-manifold in the latent space, which is realigned with the newly-generated data, hence why they appear less often in the generated images, which use samples from the prior.

Fig. 14: Data Accumulation Maintains Model Capabilities. Image reconstructions from the test set. Left: Training on prior iterations collapses the model's capability, and subsequently, it can only represent a single mode. Middle: training on aggregated data preserves model capabilities and leads to little to no degradation in the reconstructed images. Right: Baseline reconstructions after 100 training epochs on the dataset.

F Linear Regression: Replacing Data with Increasing Sample Size

In the framework of (20) and Dohmatob et al. (9), we consider sequences of linear models fit to the previous model's synthetic outputs. Within this framework, Dohmatob et al. (9) proved that if data are replaced with each model fitting iteration and the training data cardinality remains constant, then the test squared error scales linearly with the number of model fitting iterations n:

$$E_{\text{test}}^{\text{Replace}}(\hat{w}_n) = \frac{\sigma^2 d}{T-d-1} \times n \tag{11}$$

In this work, we lightly adapt the argument of Dohmatob et al. (9) to study the effects if data accumulate with each model fitting iteration. We specifically considered the case where the training data cardinality increases by a constant T with each model-fitting iteration i.e. the ith model is fit using $T \times i$ data, where T data are "real" and then each subsequently fit model contributes its own T synthetic data to the accumulating data. In this setting, the test squared error is upper bounded independent of the number of iterations.

$$E_{\text{test}}^{\text{Accumulate}}(\hat{w}_n) = \frac{\sigma^2 d}{T-d-1} \times \sum_{k=1}^{n} \frac{1}{k^2} \leq \frac{\sigma^2 d}{T-d-1} \times \frac{\pi^2}{6} \tag{12}$$

In the main text, we focus on the replace and accumulate data settings because prior work focused on replacing data and we wished to study how accumulating data affects model collapse. However, a much richer landscape of outcomes is possible. For instance, and as pointed out in personal correspondence with Dohmatob et al. (9), one can consider what we term the "Replace-Multiple" setting, in which one fits the i-th linear model using $T \times i$ data sampled from the $(i-1)$-th linear model. Replace-Multiple is a useful baseline for Accumulate because it matches the amount of training data at each model fitting iteration. Under Replace-Multiple, the test squared error grows logarithmically:

$$E_{\text{test}}^{\text{Replace-Multiple}}(\hat{w}_n) = \frac{\sigma^2 d}{T-d-1} \times \sum_{k=1}^{n} \frac{1}{k} \approx \frac{\sigma^2 d}{T-d-1} \times \log(n) \tag{13}$$

Replace-Multiple has the drawback of not matching the total amount of compute of Accumulate since each iteration of Replace-Multiple draws $T \times i$ samples from the most recent model, whereas Accumulate draws T samples from the most recent model. Other baselines are also possible, but we leave these to future work. We focus on accumulating data as we feel real and synthetic data are likely to accumulate in the real world as time progresses.

G Additional Linear Regression Numerical Results

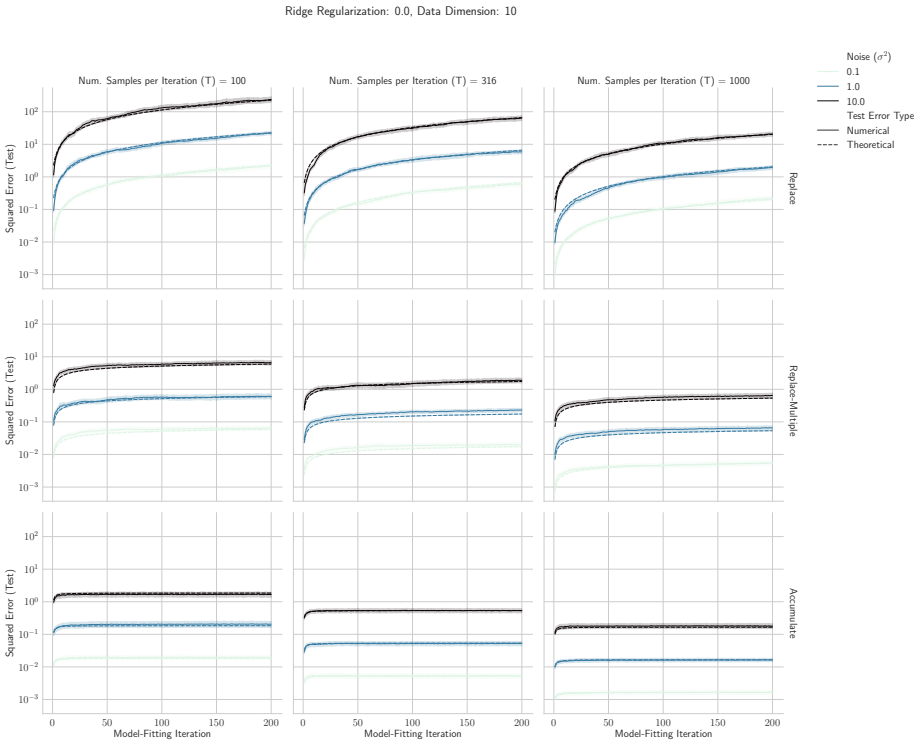

Fig. 15: Accumulating data across iterations avoids model collapse in linear regression. We consider sequences of linear models recurrently fit to generated targets by previous iterations of models. Replace (Top): If each linear model is fit to the generated targets of *only* the preceding linear model i.e. data are replaced, then the test squared error grows linearly with the number of model-fitting iterations iterations n. Replace-Multiple (Middle): If each linear model is fit to $T \times i$ samples from the $(i-1)$-th model (i.e. the same amount of data as Accumulate), then the test squared error grows logarithmically with the number of model-fitting iterations; see Appendix F for more details. Accumulate (Bottom): If each linear model is instead fit to the generate targets of *all* the preceding linear models i.e. data accumulate, then the test squared error has a finite upper bound, independent of the number of iterations. This suggests that data accumulation might be a robust solution for mitigating model collapse. This figure is similar to Figure 7 but displaying *log* test squared error and more model-fitting iterations for additional clarity.