A Closer Look at Model Collapse: From a Generalization-to-Memorization Perspective

Lianghe Shi [*1] **Meng Wu** [*1] **Huijie Zhang** [1] **Zekai Zhang** [1] **Molei Tao** [2] **Qing Qu** [1]

Abstract

This paper identifies a transition from generalization to memorization over the recursive training of diffusion models, providing a novel perspective for the study of model collapse. Specifically, the models increasingly replicate training data instead of generating novel content during iterative training on self-generated samples. This transition is directly driven by the declining entropy of the synthetic training data produced in each training cycle, which serves as a clear indicator of model degradation. Motivated by this insight, we propose an entropy-based data selection strategy to mitigate the transition from generalization to memorization and quality degradation. Empirical results show that our approach significantly enhances visual quality and diversity in recursive generation, effectively preventing model collapse.

1. Introduction

As generative models like diffusion models gain widespread use in image and video synthesis, a growing volume of synthetic content is appearing online. Given their realism—often indistinguishable from real data—future training datasets will inevitably include a substantial portion of AI-generated samples (Dohmatob et al., 2024b; Schaeffer et al., 2025). In this self-consuming loop, each iteration[1] reuses generated data to train the next-generation model. Recent studies show that even a small fraction of synthetic data can degrade performance over iterations (Dohmatob et al., 2024b), a phenomenon known as *model collapse*, which poses a serious risk to the future of generative modeling.

As surveyed by (Schaeffer et al., 2025), recent studies have identified various collapse behaviors that depend on the performance metrics employed. A series of papers (Shumailov et al., 2024; Kazdan et al., 2024; Bertrand et al., 2024; Suresh et al., 2024) reveal the model collapse phenomenon through the variance of the learned distribution. They empirically and theoretically show that the model continually loses information in the distribution tail, with variance tending towards 0. Despite significant theoretical insights regarding variance dynamics, the reduction of variance to negligible levels typically occurs only after an extremely large number of iterations. As noted by (Schaeffer et al., 2025; Kazdan et al., 2024), the collapse progresses at such a slow pace that it is rarely a practical concern in real-world applications. In contrast, the visual quality and diversity of generated images deteriorate rapidly. Another line of work (Dohmatob et al., 2024a; Gerstgrasser et al., 2024; Bertrand et al., 2024; Dohmatob et al., 2024c; Fu et al., 2024) investigates the issue from the perspective of population risk or distribution shifts. These studies observe that the generated distribution progressively deviates from the underlying distribution, causing the model's population risk to increase throughout the recursive process. Although population risk or distribution shifts offer a holistic view of performance degradation, they do not adequately characterize specific collapse behaviors.

Accordingly, this paper conducts an in-depth investigation into the collapse dynamics of diffusion models and identifies a *generalization-to-memorization* transition occurring across successive iterations, which provides a novel perspective for studying model collapse and highlights a practical issue arising from training on synthetic data. Specifically, during early iterations, the model demonstrates a strong capability to generate novel images distinct from those in the training set but gradually shifts towards memorization in later iterations, merely replicating training images. This transition significantly reduces the diversity of generated content and results in higher FID scores. Moreover, directly reproducing images from training datasets may raise copyright concerns (Ross et al., 2024; Zhang et al., 2024). Furthermore, we reveal a strong linear correlation between the generalizability of the trained model and the entropy of its training dataset, providing an empirical explanation

[*]Equal contribution [1]Department of Electrical Engineering & Computer Science, University of Michigan [2]Georgia Institute of Technology. Correspondence to: Qing Qu <qingqu@umich.edu>.

Published at ICML 2025 Workshop on the Impact of Memorization on Trustworthy Foundation Models, Vancouver, Canada. PMLR 267, 2025.

[1]In this paper, "iteration" refers to a full cycle of training and generation, not a gradient update. The pipeline illustration of the self-consuming loop can be found in Figure 5 of Appendix A.

for the generalization-to-memorization transition: as iterations progress, the entropy of the data distribution sharply decreases, directly resulting in a decline in the model's generalizability. Motivated by these empirical findings, we propose entropy-based selection methods to construct training subsets from candidate pools. Extensive experimental validation demonstrates that our proposed methods effectively identify high-entropy subsets, thereby decelerating the generalization-to-memorization transition. Additionally, our approach achieves superior image quality and lower FID scores in recursive training loops compared to the baselines.

2. Notations and Setup

In this work, we focus on the image generation task. Let $\mathcal{X}$ be the d-dimensional image space, $\mathcal{X} \subseteq \mathbb{R}^d$ and let P_0 be a data distribution over the space $\mathcal{X}$. We use **bold** letters to denote vectors in $\mathcal{X}$. We assume the original training data $\mathcal{D}_{\text{real}} = \{\boldsymbol{x}_{\text{real}}^{(1)}, \ldots, \boldsymbol{x}_{\text{real}}^{(N_0)}\}$ are generated independently and identically distributed (i.i.d.) according to P_0.

Self-Consuming Loop. Following the standard setup in model collapse studies (Alemohammad et al., 2024b; Dohmatob et al., 2024b; Kazdan et al., 2024; Dohmatob et al., 2024d;a; Alemohammad et al., 2024a; Feng et al., 2024), we denote the training dataset at the n-th iteration by $\mathcal{D}_n$. A diffusion model is trained on $\mathcal{D}_n$ using a training algorithm $\mathcal{A}(\cdot)$, which outputs a generative model associated with the distribution P_n, i.e., $P_n = \mathcal{A}(\mathcal{D}_n)$. This model is then used to sample a synthetic dataset of size N_n, denoted by $\mathcal{G}_n \sim P_n^{N_n}$, which serves as the training data for the next iterations. Based on the specific way of constructing training datasets at each iteration, previous studies (Alemohammad et al., 2024a; Dey & Donoho, 2024) distinguish different iterative paradigms:

- **The replaced training dataset.** At each iteration, the training dataset consists solely of synthetic data generated by the previous diffusion model, i.e., $\mathcal{D}_n = \mathcal{G}_{n-1}$ for $n \geq 2$ and $\mathcal{D}_1 = \mathcal{D}_{real}$. Several studies (Gerstgrasser et al., 2024; Kazdan et al., 2024; Dey & Donoho, 2024) refer to this as the "*replace*" paradigm.
- **The accumulated training dataset.** Another paradigm (Alemohammad et al., 2024a; Gerstgrasser et al., 2024) maintains access to all previous data, thereby including both real images and all synthetic images generated thus far, i.e., $\mathcal{D}_n = (\cup_{j=1}^{n-1} \mathcal{G}_j) \cup \mathcal{D}_{\text{real}}$. However, continuously increasing the training dataset size quickly demands substantial computational resources. A practical compromise is to subsample a fixed-size subset from the combined pool of candidates, referred to as the "*accumulate-subsample*" paradigm in (Kazdan et al., 2024).

This work focuses on the replace and accumulate-subsample paradigms following prior studies in (Shumailov et al., 2024; Suresh et al., 2024; Kazdan et al., 2024).

3. Model Collapses from Generalization to Memorization

In this section, we empirically demonstrate the transition from generalization to memorization that occurs over recursive iterations and investigate the underlying factors driving this transition. All experiments in this section are conducted on the CIFAR-10 dataset (Krizhevsky et al., 2009), under the *replace* paradigm, where each model is trained solely on samples generated by the model from the previous iteration.

Generalization Score. To quantify generalizability (Yoon et al., 2023; Alaa et al., 2022; Zhang et al., 2024), we adopt the *generalization score*, defined as the mean distance between each generated image and its nearest training image:

$$\texttt{GS}(n) \triangleq \texttt{Dist}(\mathcal{D}_n, \mathcal{G}_n) = \frac{1}{|\mathcal{G}_n|} \sum_{\boldsymbol{x} \in \mathcal{G}_n} \min_{\boldsymbol{z} \in \mathcal{D}_n} \kappa(\boldsymbol{x}, \boldsymbol{z}), \quad (1)$$

where $\kappa(\cdot, \cdot) : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ denotes a distance metric between two data points. A higher generalization score $\texttt{GS}(n)$ indicates that the model generates novel images rather than replicating training samples. Despite varying definitions in prior studies (Zhang et al., 2024; Alaa et al., 2022), those definitions share a common reliance on nearest neighbor.

Highlight of the Following Observations: The generated data progressively collapses into numerous compact local clusters over model collapse iterations, as evidenced by both the sharp decline in entropy over iterations and SVD visualizations. This localized concentration of data points then facilitates memorization in subsequent models, reducing their ability to generate novel images. Our claim is supported by the following three findings.

Finding I: Generalization-to-Memorization Transition. Figure 1 (Left) visualizes generated samples alongside their nearest neighbors in the training set. With a relatively large sample size, i.e., 32,768 real samples as the starting training dataset, the model exhibits strong generalizability in early iterations, producing high-quality novel images with little resemblance to training samples. However, the generalization deteriorates rapidly, and the model can only copy images from the training dataset after several iterations of training on synthetic data. Figure 1 (Right) further provides quantitative results of the generalization score that drops almost exponentially over successive iterations, providing strong evidence for the generalization-to-memorization transition. We follow the protocol of (Zhang et al., 2024) and construct six nested CIFAR-10 subsets of increasing size: $|\mathcal{D}_1| \in \{1{,}024; 2{,}048; 4{,}096; 8{,}192; 16{,}384; 32{,}768\}$. These subsets span approximately 3% to 65% of the full training set, providing controlled start points that reflect varying degrees of memorization and generalization. The

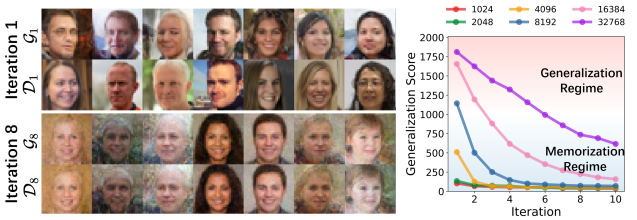
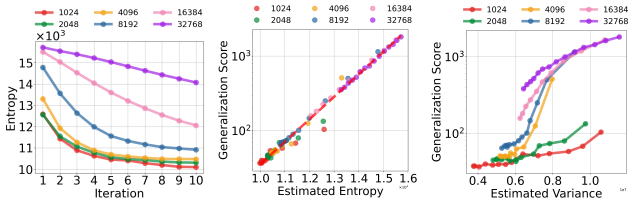


Figure 1: **The generalization-to-memorization transition. Left:** visualization of the generated images ($\mathcal{G}_n$) and their nearest neighbors in the training dataset ($\mathcal{D}_n$). **Right:** quantitative results of the generalization score of models over iterations. Different dataset sizes, in different colors.



(a) Entropy (b) GS vs. entropy (c) GS vs. variance

Figure 2: **Decreasing entropy (a) and scatter plots (b)-(c)** of the generalization score and properties of the training dataset (entropy and variance).

decline is noticeably slower for larger training subsets, indicating that larger sample sizes preserve generalization longer and delay the onset of memorization. For the smallest dataset of $1{,}024$ images, the model enters the memorization regime from the first iteration and remains there throughout.

Finding II: The Entropy of the Training Set Shrinks Sharply over Iterations. We identify entropy as the key evolving factor in the training data that drives the transition from generalization to memorization. Prior work (Yoon et al., 2023) interprets generalization in diffusion models as a failure to memorize the entire training set. (Zhang et al., 2024) further shows that diffusion models tend to generalize when trained on large datasets (e.g., $> 2^{14}$ images in CIFAR-10) and to memorize when trained on small ones (e.g., $< 2^9$ images). However, since we fix the size of the training dataset for every iteration, the previous conclusion (Zhang et al., 2024) that a larger dataset leads to generalization cannot fully explain the phenomenon observed in Finding I. We hypothesize that although sample size remains constant, the information entropy decreases over time, making it easier for the model to memorize. Based on this hypothesis, we use the following Kozachenko-Leonenko (KL) estimator (Kozachenko, 1987) to empirically estimate the *differential entropy* $H(\boldsymbol{X})$ (Cover, 1999) from a finite set $\mathcal{D}$ of samples i.i.d. drawn from the distribution P: $\hat{H}_\gamma(\mathcal{D}) = \psi(|\mathcal{D}|) - \psi(\gamma) + \log c_d + \frac{d}{|\mathcal{D}|}\sum_{\boldsymbol{x}\in\mathcal{D}} \log \varepsilon_\gamma(\boldsymbol{x})$, where $\psi : \mathbb{N} \to \mathbb{R}$ is the digamma function; γ is any positive integer; c_d denotes the volume of the unit ball in the d-dimensional space; and $\varepsilon_\gamma(\boldsymbol{x}) = \kappa(\boldsymbol{x}, \boldsymbol{x}_\gamma)$ represents the γ-nearest neighbor distance, where $\boldsymbol{x}_\gamma$ is the γ-th nearest neighbor of $\boldsymbol{x}$ in the set $\mathcal{D}$. We set $\gamma = 1$ in this paper.

As shown in Figure 2a, the entropy of the generated image dataset—used as the training set in the next iteration—consistently decreases over iterations. With a fixed dataset size, the only dataset-dependent term in the KL estimator is the sum of nearest-neighbor distances $\varepsilon(\boldsymbol{x})$, indicating that samples in $\mathcal{D}$ become increasingly concentrated and the distribution becomes spiky. Appendix E.1 presents SVD visualizations of the generated data points, illustrating that the data points locally collapse into numerous clusters.

Finding III: The Correlation between Entropy and Generalization Score. We verify that the generalization score of the trained model is strongly correlated with the entropy of the training dataset. In Figure 2b, we present a scatter plot of entropy versus generalization score across different dataset sizes and successive iterations, with the y-axis shown on a logarithmic scale. Notably, the entropy of the training dataset exhibits a significant linear relationship with the logarithm of the generalization score. Training datasets with higher entropy consistently yield better generalization in the diffusion model. The Pearson correlation coefficient is 0.91 with a p-value near zero, quantitatively confirming the strength of this correlation. Furthermore, the scatter points corresponding to different dataset sizes are all approximately aligned along a single line, suggesting the generality of the relationship between entropy and generalization. For comparison, Figure 2c shows the scatter plot of variance versus generalization score. The correlation appears substantially weaker than in Figure 2b, suggesting that variance in the training dataset may not directly influence the generalization performance of the trained model.

Conclusion. Findings I–III collectively indicate that the generated data gradually collapses into compact clusters, as shown by declining entropy. This concentration promotes memorization in the model of the next iteration and reduces their ability to generate novel images.

4. Mitigating Model Collapse with Data Selection Methods

In this section, we propose a sample selection method that selects a subset of training images from the candidate pool $\mathcal{S}$ to mitigate model collapse. Motivated by the empirical finding in Section 3, the selected training images should have high entropy. This objective can be formalized as:

$$\max_{\mathcal{D}\subset\mathcal{S},\,|\mathcal{D}|=N} \hat{H}_1(\mathcal{D}) \Leftrightarrow \max_{\mathcal{D}\subset\mathcal{S},\,|\mathcal{D}|=N} \sum_{\boldsymbol{x}\in\mathcal{D}} \log \min_{\boldsymbol{y}\in\mathcal{D}\setminus\boldsymbol{x}} \kappa(\boldsymbol{x}, \boldsymbol{y}).$$

In the accumulate-subsample setting, $\mathcal{S}$ includes all images generated so far; in the replace setting, it consists of the previous model's outputs, sized twice the target train-

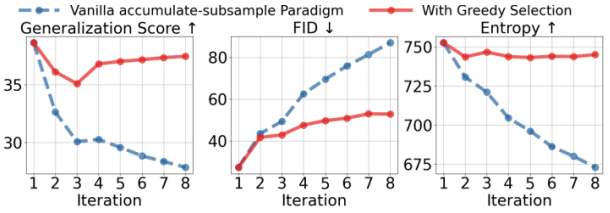


Figure 3: **Metrics over iterations. Left:** Generalization score of the trained model. **Middle:** FID of the generated images. **Right:** Estimated entropy of the training datasets.

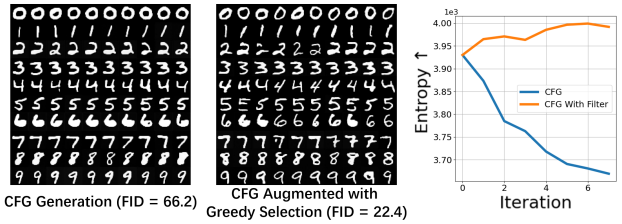


Figure 4: **Left and Middle**: Generated samples at the 8-th iteration for the vanilla CFG method and CFG augmented with Greedy Selection, respectively. **Right**: The entropy of the selected training dataset.

ing set. This non-convex max-min problem is difficult to solve globally. Instead, we propose **Greedy Selection** to approximate the solution by iteratively constructing a subset $\mathcal{D} \subset \mathcal{S}$ of size n, adding the farthest point at each step as follows: **1. Initialization:** Randomly select an initial point from the dataset $\mathcal{S}$ and add it to the set $\mathcal{D}$. **2. Iterative Selection:** At each iteration, for every candidate point $\boldsymbol{x} \in \mathcal{S} \setminus \mathcal{D}$, compute the minimum distance from $\boldsymbol{x}$ to all points currently in $\mathcal{D}$. Select the point with the *maximum* of these minimum distances and add it to $\mathcal{D}$, i.e., $\boldsymbol{x}_{select} = \arg\max_{\boldsymbol{x} \in S \setminus \mathcal{D}} \min_{\boldsymbol{y} \in \mathcal{D}} \kappa(\boldsymbol{x}, \boldsymbol{y})$. **3. Termination:** Repeat the selection process until $|\mathcal{D}| = N$.

In practice, we first extract image features using a DINOv2 (Oquab et al., 2024) model and compute distances in the feature space, i.e., $\kappa(\boldsymbol{x}, \boldsymbol{y}) = \|h(\boldsymbol{x}) - h(\boldsymbol{y})\|_2$, where $h(\cdot)$ is the feature extractor. See Appendix C.2 for the pseudocode.

5. Experiments

We empirically evaluate how Greedy Selection interacts with the *accumulate-subsample* self-consuming paradigm. Our method serves as a plug-in component for selecting high-quality and diverse training images from the candidate pool. We present some results on CIFAR-10 (Krizhevsky et al., 2009) datasets in this section. The detailed experimental setup can also be found in Appendix D. We include more results of (1) an additional data selection method, (2) additional datasets, and (3) an additional self-consuming paradigm—the replace paradigm—in Appendix F.

Results. We present the results of the generalization score, FID score, entropy, and the selected data points in Figure Figure 3. We find that Greedy Selection consistently improves generalization scores over iterations, thereby mitigating the memorization. Importantly, our selection methods can also slow down FID degradation. For example, the vanilla accumulate-subsample paradigm reaches an FID of 88.4 at iteration 8, whereas Greedy Selection significantly reduces it to 52.1. See generated images in Appendix F.3.

Analysis for the Improvement. We present the estimated entropy of the training datasets in Figure 3, which shows our selection methods effectively maintain the entropy of the training data at each iteration, consistent with their design. These more diverse, higher-entropy datasets subsequently enable the next-iteration model to generalize better, as we demonstrate in Section 3. We further investigate which samples are selected by Greedy Selection under the accumulate-subsample paradigm, where a subset of candidate data is selected for subsequent training. We show in Appendix F.2 that Greedy Selection consistently selects a significantly higher proportion of real images compared to the vanilla accumulate-subsample, thereby contributing to a lower FID.

Diversity Improvement on Classifier-Free Diffusion Guidance (CFG) (Ho & Salimans, 2022). We further validate the effectiveness of our methods in the CFG setting, showing that they substantially improve the diversity of generated images. CFG is a widely used conditional generation technique that consistently enhances perceptual quality but often sacrifices diversity (Chidambaram et al., 2024; Ho & Salimans, 2022). Prior work by (Yoon et al., 2024) identifies the CFG scale as a key factor in model collapse and suggests that a moderate scale can help mitigate it. Experimentally, Figure 4 shows that the vanilla CFG indeed generates clear MNIST images even after 8 iterations. However, the diversity collapses rapidly, with samples within each class soon becoming nearly identical. In contrast, augmenting CFG with our data selection method greatly improves image diversity and yields significantly lower FID scores compared to vanilla CFG. These results demonstrate that our approach mitigates CFG's diversity loss while preserving its quality advantage throughout the self-consuming loop.

6. Conclusion

In this work, we reveal that the memorization issue naturally arises under recursive training, highlighting a serious practical concern and offering a new perspective for studying model collapse. We empirically demonstrate that the entropy of the training data decreases over iterations and is strongly correlated with the model's generalizability. Motivated by the findings, we propose an entropy-based data selection strategy that effectively alleviates the generalization-to-memorization transition and improves image quality, thus mitigating model collapse.

Impact Statement

In this work, we investigate a critical failure mode of diffusion models known as model collapse, which occurs when models are recursively trained on synthetic data and gradually lose their generalization ability and generative diversity. As AI-generated data is unintentionally or deliberately incorporated into the training sets of next-generation models, understanding and mitigating model collapse is essential for ensuring long-term model reliability and performance. Our study identifies the generalization-to-memorization transition, demonstrates the relation between the entropy of the training set and the generalizability of the trained model, and proposes practical solutions to mitigate model collapse through entropy-based data selection.

We believe our findings will contribute to the responsible development and deployment of generative models, especially in scenarios where data sources may be mixed or partially synthetic. While the empirical findings in this paper may be misused to intentionally degrade generative models through poisoning attacks, our intent is solely to build more robust, transparent, and self-aware AI systems. We encourage researchers in generative AI to use these results to mitigate model collapse and to build reliable models, even when training data contains AI-generated samples—a scenario that may become increasingly common in the future.

Acknowledgement

LS, MW, HZ, ZZ, and QQ acknowledge funding support from NSF CCF-2212066, NSF CCF-2212326, NSF IIS 2402950, and ONR N000142512339, and Google Research Scholar Award. MT is grateful for partial supports by NSF Grants DMS-1847802, DMS-2513699, DOE Grant DE-NA0004261, Cullen-Peck Scholarship, and Emory-GT AI.Humanity Award. We also thank Prof. Peng Wang (University of Michigan and University of Macau), Mr. Xiang Li (University of Michigan), and Mr. Xiao Li (University of Michigan) for fruitful discussions and valuable feedback.

References

Abbar, S., Amer-Yahia, S., Indyk, P., Mahabadi, S., and Varadarajan, K. R. Diverse near neighbor problem. In *SoCG*, 2013.

Alaa, A. M., van Breugel, B., Saveliev, E. S., and van der Schaar, M. How faithful is your synthetic data? sample-level metrics for evaluating and auditing generative models. In *ICML*, 2022.

Alemohammad, S., Casco-Rodriguez, J., Luzi, L., Humayun, A. I., Babaei, H., LeJeune, D., Siahkoohi, A., and Baraniuk, R. G. Self-consuming generative models go MAD. In *ICML*, 2024a.

Alemohammad, S., Humayun, A. I., Agarwal, S., Collomosse, J. P., and Baraniuk, R. G. Self-improving diffusion models with synthetic data. *arxiv preprint arxiv:2408.16333*, 2024b.

An, J., Wang, D., Guo, P., Luo, J., and Schwing, A. On inductive biases that enable generalization in diffusion transformers, 2025. URL https://openreview.net/forum?id=lWGXftRS5h.

Bertrand, Q., Bose, A. J., Duplessis, A., Jiralerspong, M., and Gidel, G. On the stability of iterative retraining of generative models on their own data. In *ICLR*, 2024.

Bohacek, M. and Farid, H. Nepotistically trained generative-ai models collapse. *arxiv preprint arxiv:2311.12202*, 2023.

Carlini, N., Hayes, J., Nasr, M., Jagielski, M., Sehwag, V., Tramèr, F., Balle, B., Ippolito, D., and Wallace, E. Extracting training data from diffusion models. In *USENIX Security Symposium*, pp. 5253–5270, 2023. URL https://www.usenix.org/conference/usenixsecurity23/presentation/carlini.

Chidambaram, M., Gatmiry, K., Chen, S., Lee, H., and Lu, J. What does guidance do? A fine-grained analysis in a simple setting. In *NeurIPS*, 2024.

Cover, T. M. *Elements of information theory*. John Wiley & Sons, 1999.

Cui, H., Pehlevan, C., and Lu, Y. M. A precise asymptotic analysis of learning diffusion models: theory and insights. *arxiv preprint arxiv:2501.03937*, 2025.

Dey, A. and Donoho, D. L. Universality of the $\pi^2/6$ pathway in avoiding model collapse. *arxiv preprint arxiv:2410.22812*, 2024.

Dohmatob, E., Feng, Y., and Kempe, J. Model collapse demystified: The case of regression. In *NeurIPS*, 2024a.

Dohmatob, E., Feng, Y., and Kempe, J. Strong model collapse. *arXiv preprint arXiv:2410.04840*, 2024b.

Dohmatob, E., Feng, Y., Subramonian, A., and Kempe, J. Strong model collapse. *arxiv preprint arxiv:2410.04840*, 2024c.

Dohmatob, E., Feng, Y., Yang, P., Charton, F., and Kempe, J. A tale of tails: Model collapse as a change of scaling laws. In *ICML*, 2024d.

Feng, Y., Dohmatob, E., Yang, P., Charton, F., and Kempe, J. Beyond model collapse: Scaling up with synthesized data requires verification. *arXiv preprint arXiv:2406.07515*, 2024.

Fu, S., Zhang, S., Wang, Y., Tian, X., and Tao, D. Towards theoretical understandings of self-consuming generative models. In *ICML*, 2024.

Gerstgrasser, M., Schaeffer, R., Dey, A., Rafailov, R., Sleight, H., Hughes, J., Korbak, T., Agrawal, R., Pai, D., Gromov, A., Roberts, D. A., Yang, D., Donoho, D. L., and Koyejo, S. Is model collapse inevitable? breaking the curse of recursion by accumulating real and synthetic data. *arXiv preprint arXiv:2404.01413*, 2024.

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. C., and Bengio, Y. Generative adversarial nets. In *NeurIPS*, pp. 2672–2680, 2014.

Gu, X., Du, C., Pang, T., Li, C., Lin, M., and Wang, Y. On memorization in diffusion models, 2024. URL https://openreview.net/forum?id=9nT8ouPui8.

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., and Hochreiter, S. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *NeurIPS*, 2017.

Ho, J. and Salimans, T. Classifier-free diffusion guidance. *arxiv preprint arxiv:2207.12598*, 2022.

Ho, J., Jain, A., and Abbeel, P. Denoising diffusion probabilistic models. In *NeurIPS*, 2020.

Jain, A., Sarda, P., and Haritsa, J. R. Providing diversity in k-nearest neighbor query results. In Dai, H., Srikant, R., and Zhang, C. (eds.), *PAKDD*, 2004.

Kadkhodaie, Z., Guth, F., Simoncelli, E. P., and Mallat, S. Generalization in diffusion models arises from geometry-adaptive harmonic representations. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=ANvmVS2Yr0.

Kamb, M. and Ganguli, S. An analytic theory of creativity in convolutional diffusion models. *arXiv preprint arXiv:2412.20292*, 2024.

Kazdan, J., Schaeffer, R., Dey, A., Gerstgrasser, M., Rafailov, R., Donoho, D. L., and Koyejo, S. Collapse or thrive? perils and promises of synthetic data in a self-generating world. *arxiv preprint arxiv:2410.16713*, 2024.

Kozachenko, L. Sample estimate of the entropy of a random vector. *Problems of Information Transmission*, 23:9, 1987.

Krizhevsky, A., Hinton, G., et al. Learning multiple layers of features from tiny images. 2009.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Li, X., Dai, Y., and Qu, Q. Understanding generalizability of diffusion models requires rethinking the hidden gaussian structure. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=Sk2duBGvrK.

Niedoba, M., Zwartsenberg, B., Murphy, K., and Wood, F. Towards a mechanistic explanation of diffusion model generalization. *arXiv preprint arXiv:2411.19339*, 2024.

Oquab, M., Darcet, T., Moutakanni, T., Vo, H. V., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., Assran, M., Ballas, N., Galuba, W., Howes, R., Huang, P., Li, S., Misra, I., Rabbat, M., Sharma, V., Synnaeve, G., Xu, H., Jégou, H., Mairal, J., Labatut, P., Joulin, A., and Bojanowski, P. Dinov2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024, 2024.

Ross, B. L., Kamkari, H., Wu, T., Hosseinzadeh, R., Liu, Z., Stein, G., Cresswell, J. C., and Loaiza-Ganem, G. A geometric framework for understanding memorization in generative models. *arxiv preprint arxiv:2411.00113*, 2024.

Schaeffer, R., Kazdan, J., Arulandu, A. C., and Koyejo, S. Position: Model collapse does not mean what you think. *arXiv preprint arXiv:2503.03150*, 2025.

Schuhmann, C., Beaumont, R., Vencu, R., Gordon, C., Wightman, R., Cherti, M., Coombes, T., Katta, A., Mullis, C., Wortsman, M., Schramowski, P., Kundurthy, S., Crowson, K., Schmidt, L., Kaczmarczyk, R., and Jitsev, J. LAION-5B: an open large-scale dataset for training next generation image-text models. In *NeurIPS*, 2022.

Seddik, M. E. A., Chen, S., Hayou, S., Youssef, P., and Debbah, M. How bad is training on synthetic data? A statistical analysis of language model collapse. *arxiv preprint arxiv:2404.05090*, 2024.

Shumailov, I., Shumaylov, Z., Zhao, Y., Papernot, N., Anderson, R. J., and Gal, Y. AI models collapse when trained on recursively generated data. *Nature*, 631(8022):755–759, 2024.

Somepalli, G., Singla, V., Goldblum, M., Geiping, J., and Goldstein, T. Diffusion art or digital forgery? investigating data replication in diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 6048–6058, 2023a.

Somepalli, G., Singla, V., Goldblum, M., Geiping, J., and Goldstein, T. Understanding and mitigating copying in diffusion models. *Advances in Neural Information Processing Systems*, 36:47783–47803, 2023b.

Song, Y. and Ermon, S. Generative modeling by estimating gradients of the data distribution. In *NeurIPS*, 2019.

Suresh, A. T., Thangaraj, A., and Khandavally, A. N. K. Rate of model collapse in recursive training. *arxiv preprint arxiv:2412.17646*, 2024.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. Rethinking the inception architecture for computer vision. In *CVPR*, pp. 2818–2826, 2016.

Vincent, P. A connection between score matching and denoising autoencoders. *Neural Computing*, 23:1661–1674, 2011.

von Platen, P., Patil, S., Lozhkov, A., Cuenca, P., Lambert, N., Rasul, K., Davaadorj, M., Nair, D., Paul, S., Berman, W., Xu, Y., Liu, S., and Wolf, T. Diffusers: State-of-the-art diffusion models. https://github.com/huggingface/diffusers, 2022.

Wang, B. and Vastola, J. J. The unreasonable effectiveness of gaussian score approximation for diffusion models and its applications. *arXiv preprint arXiv:2412.09726*, 2024.

Wang, P., Zhang, H., Zhang, Z., Chen, S., Ma, Y., and Qu, Q. Diffusion models learn low-dimensional distributions via subspace clustering. *arXiv preprint arXiv:2409.02426*, 2024a.

Wang, Y., He, Y., and Tao, M. Evaluating the design space of diffusion-based generative models. *NeurIPS*, 2024b.

Wen, Y., Liu, Y., Chen, C., and Lyu, L. Detecting, explaining, and mitigating memorization in diffusion models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=84n3UwkH7b.

Wyllie, S. C., Shumailov, I., and Papernot, N. Fairness feedback loops: Training on synthetic data amplifies bias. In *ACM FAccT*, pp. 2113–2147, 2024.

Yoon, T., Choi, J. Y., Kwon, S., and Ryu, E. K. Diffusion probabilistic models generalize when they fail to memorize. In *workshop on structured probabilistic inference {\&} generative modeling, ICML*, 2023.

Yoon, Y., Hu, D., Weissburg, I., Qin, Y., and Jeong, H. Model collapse in the self-consuming chain of diffusion finetuning: A novel perspective from quantitative trait modeling. *arXiv preprint arXiv:2407.17493*, 2024.

Zhang, H., Zhou, J., Lu, Y., Guo, M., Wang, P., Shen, L., and Qu, Q. The emergence of reproducibility and consistency in diffusion models. In *ICML*, 2024.

Zhang, X., Wei, X., Wu, J., Wu, J., Zhang, Z., Lei, Z., and Li, Q. Generating on generated: An approach towards self-evolving diffusion models. *arxiv preprint arxiv:2502.09963*, 2025.

Zhu, X., Cheng, D., Li, H., Zhang, K., Hua, E., Lv, X., Ding, N., Lin, Z., Zheng, Z., and Zhou, B. How to synthesize text data without model collapse? *arxiv preprint arxiv:2412.14689*, 2024.

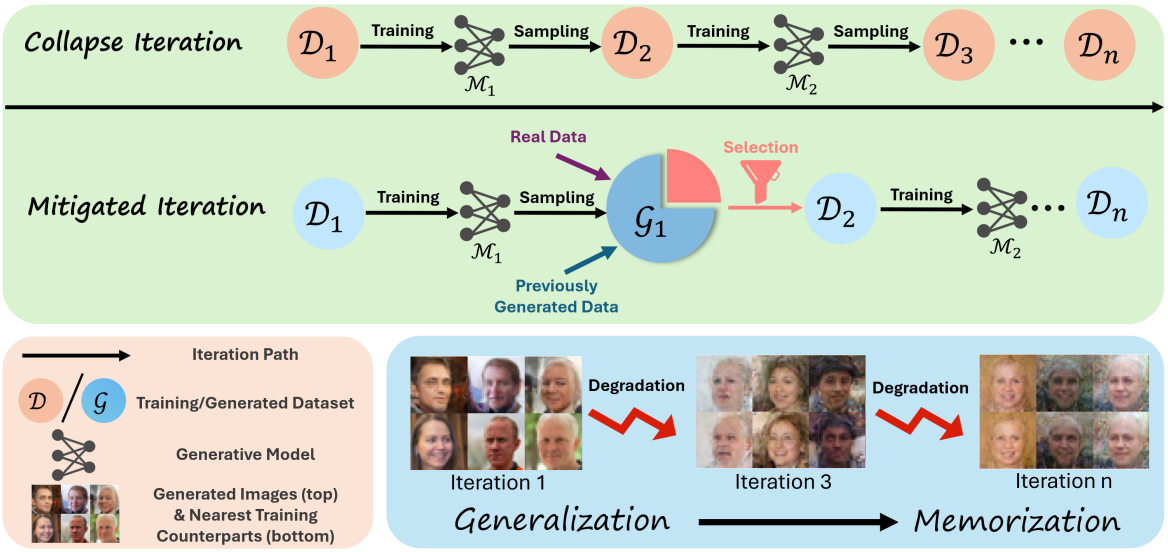


Figure 5: **High-level depiction of the self-consuming pipeline. Top:** *Collapse iteration* represents the replace paradigm where models are trained solely on synthetic images generated by the previous diffusion model. **Middle:** In the *mitigated iteration*, original real data and previously generated data are added to train the next-generation model. Our proposed selection methods construct a training subset and can further mitigate collapse. **Bottom Right:** Evolution of the generated images.

A. Background

A.1. Self-Consuming Loop

We show the recursive process of the self-consuming training loop in Figure 5.

Based on the specific way of constructing training datasets, previous studies (Alemohammad et al., 2024a; Kazdan et al., 2024; Dey & Donoho, 2024) distinguish two distinct iterative paradigms:

- **The replaced training dataset.** At each iteration, the training dataset consists solely of synthetic data generated by the previous diffusion model, i.e., $\mathcal{D}_n = \mathcal{G}_{n-1}$. Several studies (Gerstgrasser et al., 2024; Kazdan et al., 2024; Dey & Donoho, 2024) refer to this as the "*replace*" paradigm and have demonstrated that under this setting, the variance collapses to 0 or the population risk diverges to infinity. We show this paradigm in the top of Figure 5, i.e., the collapse iteration.

- **The accumulated training dataset.** A more realistic paradigm (Alemohammad et al., 2024a; Gerstgrasser et al., 2024) is to maintain access to all previous data, thereby including both real images and all synthetic images generated thus far, i.e., $\mathcal{D}_n = (\cup_{j=1}^{n-1}\mathcal{G}_j) \cup \mathcal{D}_{\text{real}}$. However, continuously increasing the training dataset size quickly demands substantial computational resources. A practical compromise is to subsample a fixed-size subset from all candidate images, referred to as the "*accumulate-subsample*" paradigm in (Kazdan et al., 2024). Under certain conditions, prior work (Gerstgrasser et al., 2024; Kazdan et al., 2024) has shown that accumulating real and synthetic data mitigates model degradation, preventing population risk from diverging. We present this accumulate-subsample paradigm augmented with our selection method in the middle of Figure 5, i.e., the mitigated iteration.

A.2. Diffusion Models

Diffusion models are a class of generative models that synthesize data by reversing a gradual noising process, achieving state-of-the-art results in image and video generation. For a given data distribution, diffusion models do not directly learn the probability density function (pdf) of the distribution; instead, they define a forward process and a reverse process, and learn the score function utilized in the reverse process. Specifically, the forward process (Ho et al., 2020) progressively adds Gaussian noise to the image, and the conditional distribution of the noisy image is given by:

$$p(\boldsymbol{x_t}|\boldsymbol{x_0}) = \mathcal{N}\left(\boldsymbol{x_t}; \sqrt{\bar{\alpha}_t}\boldsymbol{x_0}, (1-\bar{\alpha}_t)\boldsymbol{I}\right),$$

where $\bar{\alpha}_t$ is the scale schedule, $\boldsymbol{x_0}$ is the clean image drawn from P_0, and $\boldsymbol{x_t}$ is the noisy image. This forward can also be described as a stochastic differential equation (SDE) (Song & Ermon, 2019):

$$d\boldsymbol{x} = f(\boldsymbol{x}, t)dt + g(t)d\boldsymbol{w}, \tag{2}$$

where $f(\cdot, t) : \mathbb{R}^d \to \mathbb{R}^d$ denotes the vector-valued drift coefficient, $g(t) \in \mathbb{R}$ is diffusion coefficient, and w is a standard Brownian motion. This SDE has a corresponding reverse SDE as

$$d\boldsymbol{x} = [f(\boldsymbol{x}, t) - g^2(t)\nabla_{\boldsymbol{x}} \log q_t]dt + g(t)d\boldsymbol{w}, \tag{3}$$

where dt represents a negative infinitesimal time step, driving the process from $t = T$ to $t = 0$. The reverse SDE enables us to gradually convert a Gaussian noise to a clean image $\boldsymbol{x} \sim P_0$.

The score function $\nabla_{\boldsymbol{x}} \log p_t$ is typically unknown and needs to be estimated using a neural network $\boldsymbol{s}_\theta(\boldsymbol{x}, t)$. The training objective can be formalized as

$$\mathbb{E}_{t\sim\mathcal{U}(0,T)}\mathbb{E}_{p_t(\boldsymbol{x})}\left[\lambda(t)\left\|\nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x}) - \boldsymbol{s}_\theta(\boldsymbol{x}, t)\right\|_2^2\right],$$

and can be efficiently optimized with score match methods such as denoising score matching (Vincent, 2011).

B. Related Work

B.1. Model Collapse

As state-of-the-art generative models continually generate better and better image quality, the AI-generated images become hard to distinguish and are inevitably mixed in the training dataset of the next-generation models. In fact, the authors of (Alemohammad et al., 2024a) show that the LAION-5B dataset (Schuhmann et al., 2022) used to train Stable Diffusion indeed contains synthetic data. Unfortunately, recent works (Alemohammad et al., 2024b; Dohmatob et al., 2024b; Kazdan et al., 2024; Dohmatob et al., 2024d;a; Alemohammad et al., 2024a; Feng et al., 2024) show that the performance of the model will degrade, and the model may eventually collapse to generate homogeneous and meaningless content. The work of (Shumailov et al., 2024) first proposes the concept of model collapse and provides a theoretical analysis framework based on the Gaussian model. Their results prove that if the Gaussian model is recurrently estimated based on the data generated by the previous model, then the variance of the Gaussian model will tend towards zero, which means the distribution finally degenerates into a delta distribution with density on only one point.

Following this important work, there has been significant research effort contributing to the **model collapse phenomenon**. (Suresh et al., 2024) extends the analysis on the Gaussian model to the Bernoulli model and Poisson model. (Dohmatob et al., 2024a) analyzes the recursive training process in the high-dimensional linear and ridge regression setting. They provide a linear error rate in their setup and investigate how to mitigate model collapse using optimal regularization. (Dohmatob et al., 2024d) claims that the scaling law for foundation models will change if we consider a training dataset incorporating synthetic data. (Wyllie et al., 2024) demonstrates the bias of the model is amplified through the self-consuming loop and further provides an algorithmic reparation method to erase the bias and negative shifts of the generated content. The work of (Seddik et al., 2024) proposes a statistical model to characterize the process of model collapse in language models and theoretically estimates the maximal proportion of synthetic data that can be incorporated into the training set without inducing collapse, supported by empirical validation. (Cui et al., 2025) theoretically investigates the model collapse in diffusion models. They consider the training dynamic of a two-layer auto-encoder, which is optimized by stochastic gradient descent. Their results show how the neural network architecture influences the generated density and how it shapes the collapse behavior.

To mitigate model collapse, several useful strategies have been proposed by prior work. One of the methods is to incorporate all samples generated in previous iterations and also the real data into the training dataset, i.e., the accumulate paradigm we refer to in the main paper. A line of studies both empirically and theoretically justifies the effectiveness of this strategy. (Gerstgrasser et al., 2024) demonstrates that the accumulate paradigm avoids model collapse; under a linear regression setup, the test error of the model does not diverge but is rather upper bounded. Then, the study of (Dey & Donoho, 2024) further proves the universality of the error upper bound across a large family of canonical statistical models. (Kazdan et al., 2024) investigates an accumulate-subsample setting, where the model is trained on a subset of the accumulated samples. They validate that the test error also plateaus and study the interaction between real and synthetic data. (Bertrand et al., 2024)

theoretically proves the stability of iterative training under two conditions: the generative model initially trained on real data is good enough, and the proportion of clean data in each iteration's training set is large enough. However, a recent paper (Dohmatob et al., 2024c) establishes a robust negative result that shows that model collapse generally persists even when mixing real and synthetic data. (Alemohammad et al., 2024a) also points out that the accumulate paradigm only delays but does not prevent model collapse. To prevent model collapse, we could incorporate fresh real data in each iteration. Other effectiveness methods include verification (Feng et al., 2024) and re-editing (Zhu et al., 2024). For instance, (Feng et al., 2024) theoretically validates the necessity of data selection. Although they demonstrate how selection can prevent model collapse, they do not propose new selection methods. (Zhu et al., 2024) focuses on language models and proposes a token editing method to prevent model collapse, with a theoretical guarantee. Compared to prior work, this paper proposes a novel entropy-based data selection method for diffusion models that improves both generalizability and image quality, thereby mitigating model collapse.

Over the last few years, numerous research papers have **studied model collapse from different perspectives and dimensions**. An important study (Schaeffer et al., 2025) makes a thorough survey about different definitions and patterns investigated in previous work. A series of papers (Shumailov et al., 2024; Kazdan et al., 2024; Bertrand et al., 2024; Suresh et al., 2024) focuses on variance collapse of the learned distribution. They empirically and theoretically show that the model continually loses information in the distribution tail, with variance tending towards 0. Another line of work (Dohmatob et al., 2024a; Gerstgrasser et al., 2024; Bertrand et al., 2024; Dohmatob et al., 2024c; Fu et al., 2024) investigates the issue from the perspective of population risk or distribution shifts. These studies observe that the generated distribution progressively deviates from the underlying distribution, causing the model's population risk to increase throughout the recursive process. Numerous studies (Alemohammad et al., 2024a; Bohacek & Farid, 2023; Zhang et al., 2025) also report that models begin generating hallucinated data. Additionally, (Dohmatob et al., 2024d) claims that synthetic data changes the scaling law. (Cui et al., 2025) studies the mode collapse (Goodfellow et al., 2014) of diffusion models over iterations, i.e., the generated distribution progressively loses modes. In this paper, we provide a novel perspective for studying model collapse in diffusion models, revealing the generalization-to-memorization transition. We then identify the reason for the transition by showing the relationship between the entropy of the training dataset and the generalizability of diffusion models. Our analysis further motivates effective data selection methods based on entropy.

B.2. Generalization and Memorization

Recent studies (Yoon et al., 2023; Zhang et al., 2024) have identified two distinct learning regimes in diffusion models, depending on the size of the training dataset and the model's capacity: (1) Memorization regime, when models with sufficient capacity are trained on limited datasets, they tend to memorize the training data; and (2) Generalization regime, as the number of training samples increases, the model begins to approximate the underlying data distribution and generate novel samples. To investigate the transition between these regimes, (Wang et al., 2024a) shows that the number of training samples required for the transition from memorization to generalization scales linearly with the intrinsic dimension of the dataset. In addition, the analysis of training and generation accuracies in (Wang et al., 2024b) provides a potential step toward quantifying generalization. Meanwhile, concurrent work also explores memorization and generalization separately. To understand generalization, studies such as (Kadkhodaie et al., 2024; An et al., 2025) attribute the generalization to implicit bias introduced by network architectures. Other works studies the generalized distribution using Gaussian models (Wang & Vastola, 2024; Li et al., 2024) and patch-wise optimal score functions (Niedoba et al., 2024; Kamb & Ganguli, 2024). As for memorization, it is investigated in both unconditional and conditional (Gu et al., 2024), as well as the text-to-image diffusion models (Carlini et al., 2023; Somepalli et al., 2023a). Additionally, methods to mitigate memorization in diffusion models have been proposed in (Somepalli et al., 2023b; Wen et al., 2024). Distinct from prior work, our study is the first to establish a connection between model collapse and the transition from generalization to memorization. This connection not only offers a novel perspective to understand model collapse but also provides insights to mitigate it by mitigating memorization.

C. Algorithm Details

C.1. Threshold Decay Filter

In this section, we introduce another data selection method—Threshold Decay Filter. The Greedy Selection can efficiently and effectively extract a subset with a large entropy. However, this greedy method carries a risk of over-optimization, which may lead to an excessively expanded distribution and a progressively increasing variance in the selected samples. To mitigate this, we also provide the following Threshold Decay Filter, which can control the filtration strength by a decaying threshold.

Algorithm 1 Greedy Selection

Input: Dataset $\mathcal{S}$, target size N, distance function $\kappa(\cdot, \cdot)$
Output: Selected subset $\mathcal{D}$ of size N
Initialize $\mathcal{D} \leftarrow \{\text{random point from } \mathcal{S}\}$
while $|\mathcal{D}| < N$ **do**
  for each $x \in \mathcal{S} \setminus \mathcal{D}$ **do**
    Compute $d(x) \leftarrow \min_{y \in \mathcal{D}} \kappa(x, y)$
  end for
  $x_{\text{select}} \leftarrow \arg\max_{x \in \mathcal{S} \setminus \mathcal{D}} d(x)$
  $\mathcal{D} \leftarrow \mathcal{D} \cup \{x_{\text{select}}\}$
end while
return $\mathcal{D}$

Threshold Decay Filter. The procedure constructs a subset $\mathcal{D} \subset \mathcal{S}$ of size N by iteratively selecting samples that are sufficiently distant from the current set $\mathcal{D}$. The algorithm proceeds as follows:

1. **Initialization:** Set an initial threshold $\tau > 0$. Randomly select one point from the dataset $\mathcal{S}$ and add it to the set $\mathcal{D}$.
2. **Threshold-based Selection:** For each point $\boldsymbol{x} \in \mathcal{S} \setminus \mathcal{D}$, compute the distance from $\boldsymbol{x}$ to all points in $\mathcal{D}$. If all distances are greater than the current threshold τ, add $\boldsymbol{x}$ to $\mathcal{D}$.
3. **Threshold Decay:** If $|\mathcal{D}| < N$ after a complete pass through $\mathcal{S} \setminus \mathcal{D}$, reduce the threshold τ by multiplying it with a decay factor $\alpha \in (0, 1)$, and repeat Step 2.
4. **Termination:** Repeat Steps 2–3 until $|\mathcal{D}| = N$.

Threshold Decay Filter is a soft variant of Greedy Selection that provides adjustable control over the selection strength. When the initial threshold is set sufficiently high and the decay factor is close to 1, the Threshold Decay Filter behaves similarly to Greedy Selection. Conversely, if both the initial threshold and decay factor are set to 0, the filter does not filter out any data point and reduces to the vanilla replace or accumulate-subsample paradigm.

There are also other possible data selection strategies to approximately optimize the entropy. However, finding the best data selection method is beyond the scope of this paper and is an interesting question for future work. The goal of this paper is to reveal the relationship between generalizability and the entropy, and validate the effectiveness of optimizing the entropy of the training dataset via data selection.

C.2. Pseudocode for the Algorithms

We present the pseudocode for Greedy Selection and Threshold Decay Filter in Algorithms 1 and 2, respectively.

D. Detailed Experimental Setup

In this section, we provide a detailed experimental setup for the experiments in this paper.

D.1. Datasets

We conduct experiments on three widely used image generation benchmarks. **CIFAR-10** (Krizhevsky et al., 2009) consists of 32×32 color images in 10 classes. Due to computational constraints, we use a subset of 32,768 training images. Our goal is not to achieve state-of-the-art FID among large diffusion models but rather to demonstrate that our method mitigates memorization in the self-consuming loop. As shown in Section 3, this subset is sufficient to observe the transition from generalization to memorization. We also conduct experiments on subsets of **FFHQ** (Jain et al., 2004), downsampled to 32×32 resolution, and **MNIST** (LeCun et al., 1998), using 8,192 and 12,000 training images, respectively.

D.2. Network Structure

For CIFAR-10 and FFHQ, we use a UNet-based backbone, taking RGB images as inputs and predicting noise residuals. Our implementation is based on the Hugging Face Diffusers base code (von Platen et al., 2022). The architecture hyperparameters

Algorithm 2 Threshold Decay Filter

Input: Dataset $\mathcal{S}$, target size N, initial threshold $\tau > 0$, decay factor $\alpha \in (0, 1)$, distance function $\kappa(\cdot, \cdot)$
Output: Selected subset $\mathcal{D}$ of size N
Initialize $\mathcal{D} \leftarrow \{\text{random point from } \mathcal{S}\}$
while $|\mathcal{D}| < N$ **do**
  added $\leftarrow$ false
  for each $x \in \mathcal{S} \setminus \mathcal{D}$ **do**
    Compute $d_{\min}(x) \leftarrow \min_{y \in \mathcal{D}} \kappa(x, y)$
    if $d_{\min}(x) > \tau$ **then**
      $\mathcal{D} \leftarrow \mathcal{D} \cup \{x\}$
      added $\leftarrow$ true
    end if
    if $|\mathcal{D}| = N$ **then**
      return $\mathcal{D}$
    end if
  end for
  if added = false **then**
    $\tau \leftarrow \alpha \cdot \tau$ // No point added $\Rightarrow$ decay threshold
  end if
end while
return $\mathcal{D}$

of the neural network are listed as follows:

- The numbers of in-channel and out-channel are 3.
- The number of groups for group normalization within Transformer blocks is 16.
- The number of layers per block is 2.
- The network contains 6 down-sampling blocks and 6 up-sampling blocks.
- The numbers of feature channels for the 6 blocks are $48, 48, 96, 96, 144, 144$ respectively.

For MNIST, we use a similar UNet-based backbone, taking single-channel images as inputs and predicting noise residuals. The architecture hyperparameters of the network are listed as follows:

- The numbers of in-channel and out-channel are 1.
- The number of groups for group normalization within Transformer blocks is 32.
- The number of layers per block is 2.
- The network contains 4 down-sampling blocks and 4 up-sampling blocks.
- The numbers of feature channels for the 4 blocks are $64, 128, 256, 512$ respectively.

D.3. Implementation Details

In this paper, we use DDPM as our generative method. For efficiency, we use the FP-16 mixing precision to train our models, which is inherently implemented by the Hugging Face Diffusers codebase. The batch size of training all datasets is set to be 128. A 1000-step denoising process is used as the reverse process. For CIFAR-10, the epoch number is set to be 500; for FFHQ, the epoch number is set to be 1000. We use the Adam optimizer with a learning rate of 10^{-4} and a weight decay of 10^{-6}. Other experimental hyperparameters are exactly the default values in the original Hugging Face Diffusers codebase. We use the DINOv2 model (Oquab et al., 2024) to extract features of images and then calculate the distance between two images in the feature space. We use the InceptionV3 model (Szegedy et al., 2016) to extract features of images to calculate the FID score. All experiments are conducted on a single NVIDIA A-100 GPU.

Figure 6: 2-D projection of data points onto the first two singular bases of the real dataset. The orange points represent the generated images at the 1-st and 21-st iterations, respectively.

D.4. Evaluation Metrics

We use the generalization score and entropy to evaluate the effectiveness of our method in mitigating memorization. Additionally, we adopt the Fréchet Inception Distance (FID) (Heusel et al., 2017) as a metric to quantify the distributional divergence between generated images and real images.

E. Addition Results for Section 3

E.1. Generated Image Distribution Becomes Spiky

In the main paper, we show the entropy of the generated dataset decreases over iterations, indicating the data distribution becomes spiky and the samples become increasingly concentrated. To visualize the collapse trend, we present the SVD visualizations of the generated data points in Figure 6, which illustrates that the data points locally form numerous clusters.

To further quantitatively measure the spiky degree of the empirical image distribution, we also adopt the Mean Nearest Neighbor Distance (MNND) (Jain et al., 2004; Abbar et al., 2013), which removes the data-independent constants and logarithm in the KL estimator of entropy:

$$\texttt{MNND}(\mathcal{D}_t) \triangleq \texttt{Dist}(\mathcal{D}_t, \mathcal{D}_t) = \frac{1}{|\mathcal{D}_t|} \sum_{\boldsymbol{x} \in \mathcal{D}_t} \min_{\boldsymbol{z} \in \mathcal{D}_t \setminus \boldsymbol{x}} d(\boldsymbol{x}, \boldsymbol{z}). \tag{4}$$

A lower MNND suggests a tightly clustered and spiky distribution, while a higher distance suggests dispersion and diversity. The KL estimator can be related to MNND through

$$e^{\frac{\hat{H}_1(\mathcal{D}_t) - B}{d}} \leq \text{MNND}(\mathcal{D}_t), \tag{5}$$

where $\hat{H}_1(\mathcal{D}_t)$ represents the estimated entropy of $\mathcal{D}_t$ with $k = 1$ and B is a constant offset given a fixed dataset size.

We want to clarify the nuance between MNND and variance. A spiky distribution does not imply that all data points are concentrated in a single small region—this behavior has already been illustrated by the variance collapse behavior (Shumailov et al., 2024; Kazdan et al., 2024; Bertrand et al., 2024). Specifically, Figure 7 shows that the variance of the generated dataset only slightly decreases along the successive iterations and is far from complete collapse. On the contrary, MNND decreases almost exponentially and reaches a small value after 10 iterations. Thus, the results align with the prior claim in (Schaeffer et al., 2025; Kazdan et al., 2024) that the collapse of variance progresses at such a slow pace that it is rarely a practical concern in real-world applications. In contrast, the collapse in MNND emerges at an early stage, highlighting a critical memorization issue caused by training on synthetic data.

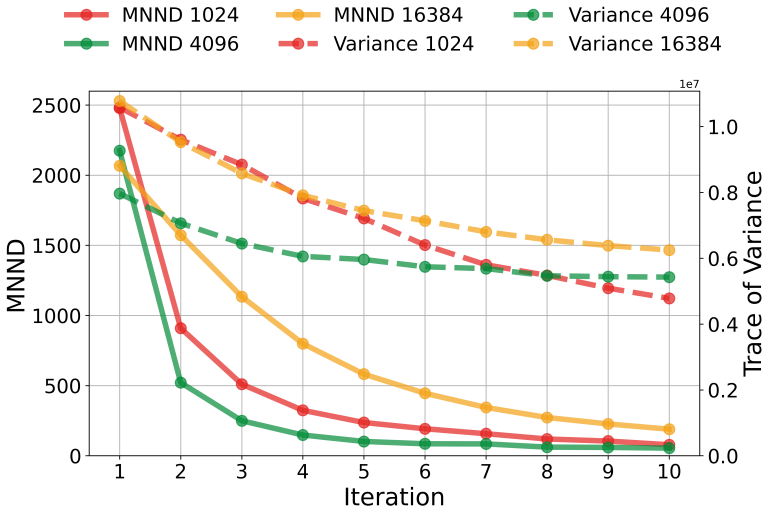


Figure 7: The MNND and the trace of the covariance matrix over iterations.

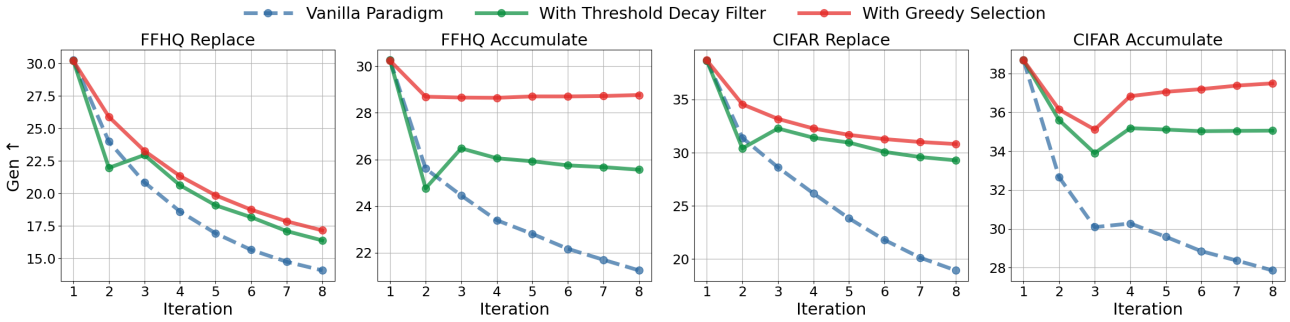


Figure 8: **Generalization Score of the trained model over iterations.** We indicate the settings on top of the subfigures. Here, "Accumulate" denotes "Accumulate-subsample". In each subfigure, three different lines are used to represent the vanilla paradigm and its variants augmented with the proposed selection methods.

F. Additional Results for Section 5

In this section, we provide additional results on the Threshold Decay Filter introduced in Appendix C.1 and also incorporate more datasets, i.e., FFHQ and MNIST. The results demonstrate that the proposed data selection strategies effectively alleviate memorization and reduce FID scores, thereby mitigating model collapse. Additionally, experiments on classifier-free guidance (CFG) (Ho & Salimans, 2022) generation show that our method effectively mitigates diversity collapse of CFG.

F.1. Detailed Results

We present the generalization score of models on CIFAR-10 and FFHQ in Figure 8, including two data selection methods—Greedy Selection and Threshold Decay Filter. Greedy Selection consistently improves generalization scores across all datasets and paradigms and is particularly effective in the accumulate-subsample setting.

Figure 9 reports the FID of generated images over successive iterations. Under the accumulate-subsample paradigm, both methods yield notable improvements in FID, with Greedy Selection outperforming Threshold Decay Filter. For example, the vanilla accumulate paradigm reaches an FID of 75.7 at iteration 8, whereas Greedy Selection significantly reduces it to 44.7. On the FFHQ dataset under the replace paradigm, however, the Threshold Decay Filter performs better, suggesting that adaptive selection strength may be beneficial in certain cases.

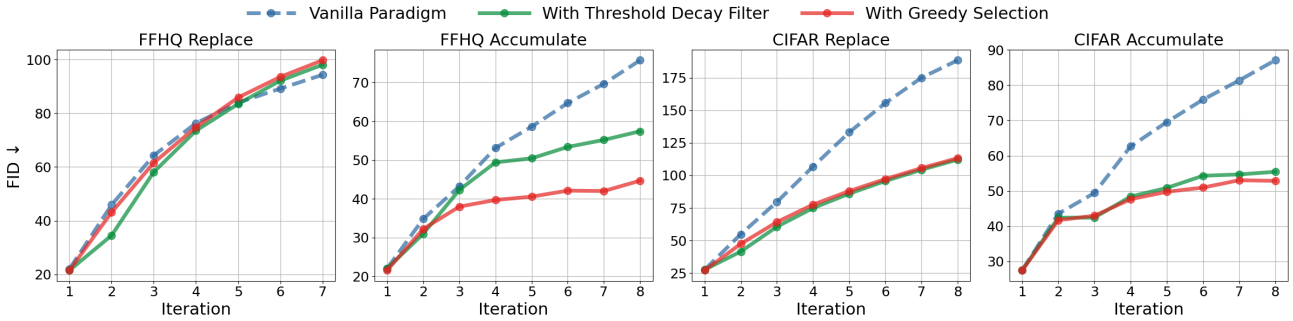


Figure 9: **FID of the generated images over iterations.** We indicate the settings on top of the subfigures. Here, "Accumulate" denotes "Accumulate-subsample". In each subfigure, three different lines are used to represent the vanilla paradigm and its variants augmented with the proposed selection methods.

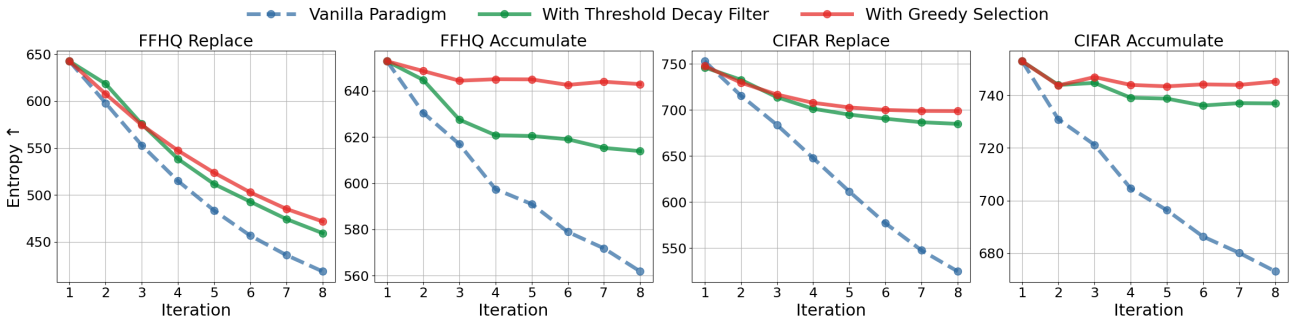


Figure 10: **Estimated entropy of the training datasets over iterations.** We indicate the settings on top of the subfigures. Here, "Accumulate" denotes "Accumulate-subsample". In each subfigure, three different lines are used to represent the vanilla paradigm and its variants augmented with our selection methods.

F.2. Improvement Analysis

We present the estimated entropy of the training datasets in Figure 10. As shown in the figure, our selection methods effectively increase the entropy of the training data at each iteration, consistent with their design. These more diverse, higher-entropy datasets subsequently enable the next-iteration model to generalize better, as evidenced by the improved generalization scores in Figure 8.

We further investigate which samples are selected by our methods under the accumulate-subsample paradigm, where the model has access to all prior synthetic images and the real images but is trained on a subset of them. Figure 11 shows the proportion of selected samples originating from different sources, with the blue bar indicating the proportion of real images. As illustrated in the figure, both Greedy Selection and Threshold Decay Filter consistently select a significantly higher proportion of real images compared to the $1/n$ reference curve. For example, on the FFHQ dataset, Greedy Selection selects 65% real images at the 8-th iteration, while vanilla accumulate-subsample includes only 12.5%. This outcome arises because the image distribution progressively collapses into compact clusters over iterations, and our selection methods tend to prioritize boundary samples—namely, real images—by maximizing the nearest neighbor distance.

F.3. Visualization of the Generalization Improvement for Section 5

We provide visualization of the generated FFHQ images with their nearest training neighbors to show the model's generalizability. Figure 12 shows some images for different training paradigms at 5-th iteration in grid format. With augmentation from the Greedy Selection method, the model generates images that deviate more from the training set compared to the vanilla accumulate paradigm, indicating a better generalization ability.

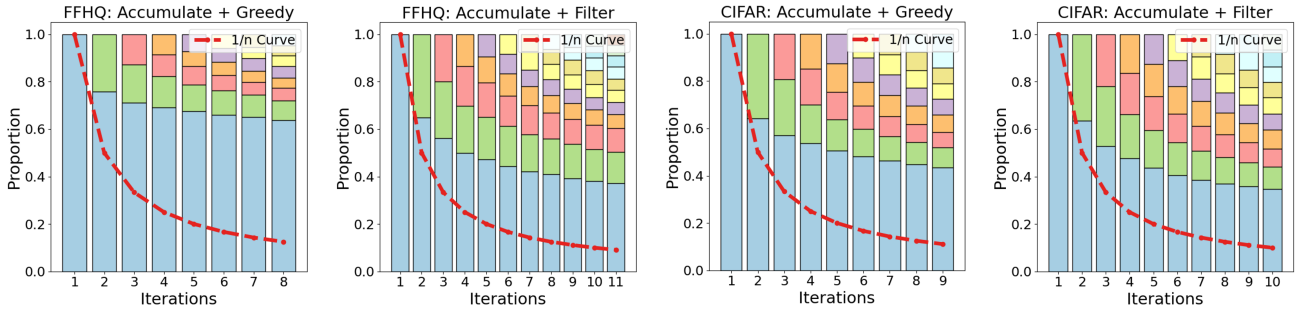


Figure 11: **Proportion of the selected images** from previous iterations or the real dataset. We use different colors to represent different sources. Particularly, the blue bars denote the proportion of the real images. The red line represents the $1/n$ curve that indicates the proportion of the real images if we evenly select the data subset from all available images (accumulate-subsample). We indicate the settings on top of the subfigures. Here, "Accumulate" denotes "Accumulate-subsample".

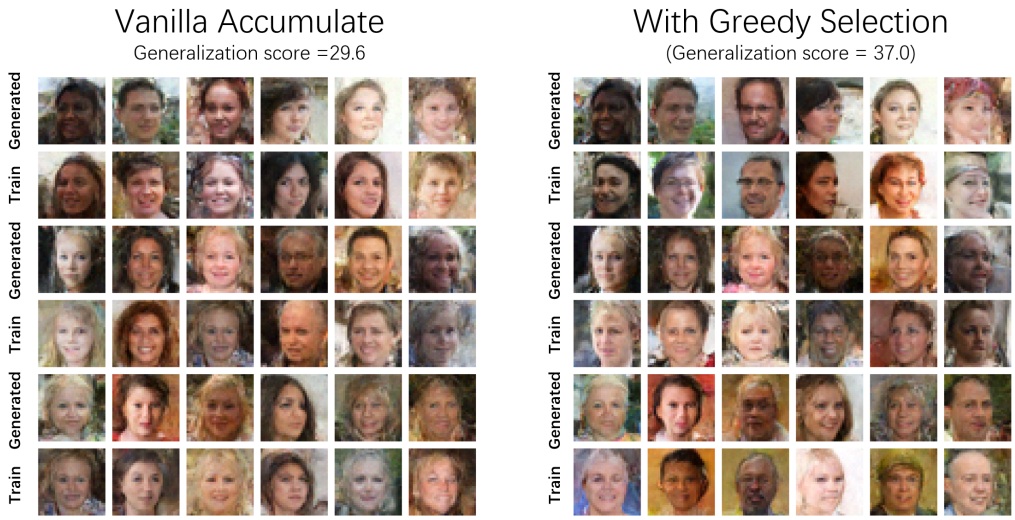


Figure 12: The visualization of the generated images and their nearest neighbors in the training dataset. Each pair of rows corresponds to one group: the top row shows the generated images, and the bottom row shows their nearest training images.

G. Ablation Study

G.1. Training on More Samples

In Section 5 of the main paper, we augment the vanilla replace paradigm with our data selection methods. Specifically, under the replace setting, the vanilla baseline generates N images and trains the next-iteration model based solely on these N images from the previous iteration. Instead, the data selection methods generate $2N$ images at each iteration and select a subset of N images from the $2N$ images for training the next-iteration model. One nature question is whether incorporating all the generated $2N$ images to train the next-iteration model can obtain a better performance.

Next, we show that when training on the whole $2N$ dataset, the performance (FID score) of the model is between the vanilla replace setting (generating N images and training on those N images) and the data selection method (generating $2N$ images and training on the selected N-images subset). The results on CIFAR-10 are shown in Figure 13.

Several conclusions can be drawn by comparing the results across these settings.

- Incorporating more data into the training-sampling recursion can indeed mitigate the rate of model collapse. Compared to the vanilla replacement paradigm (i.e., the first line), using $2N$ images (second line) yields improved performance. This aligns with prior findings (Kazdan et al., 2024; Fu et al., 2024) that sample size is a key factor influencing the collapse rate.

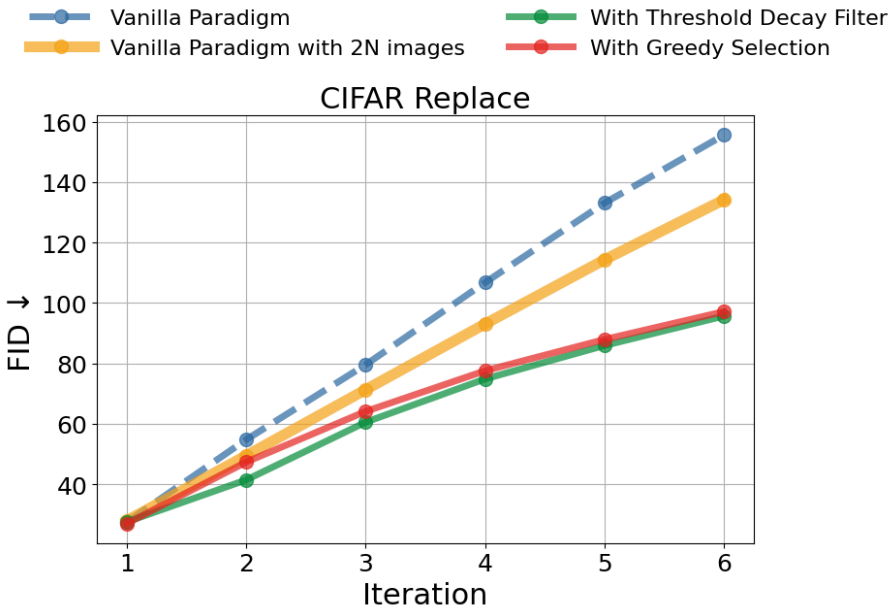


Figure 13: **FID score of the generated images over iterations.** We add an additional setting of generating $2N$ images and training the next model on the entire $2N$-images data without filtering (the line in yellow).

- Further filtering the training data with our selection method leads to even better performance than training on the full set of $2N$ images. The results validate the effectiveness of our selection method, achieving a lower FID score while largely decreasing the training budget.

G.2. Different Decay Rates

The decay rate is one important hyperparameter for the Threshold Decay Filter. In this section, we use different decay rates and show that the filter is robust to a wide range of decay rates.

Figure 14 shows the FID scores of generated CIFAR-10 images across iterations for different methods. As shown in the figure, the data selection methods consistently outperform the vanilla accumulate paradigm, indicating strong robustness of the hyperparameter.

H. Generated Images over Iterations

We provide additional generated images of trained models across iterations and dataset sizes to show the collapse process.

Figures 15 to 20 present the generated images of the model trained on CIFAR-10 with various dataset sizes across successive iterations.

As shown in Figures 15 and 16, the diffusion models tend to memorize small training datasets throughout the recursive process, exactly copying training images during generation. Because the generated samples are nearly identical to the training data, image quality does not noticeably degrade. However, duplicated generations emerge in later iterations, and diversity declines as the model gradually loses coverage over parts of the original images.

On the contrary, on the large dataset of $32{,}768$ images, the models generate novel images in the beginning. As the model collapses, the quality and diversity of the images gradually degrade over iterations.

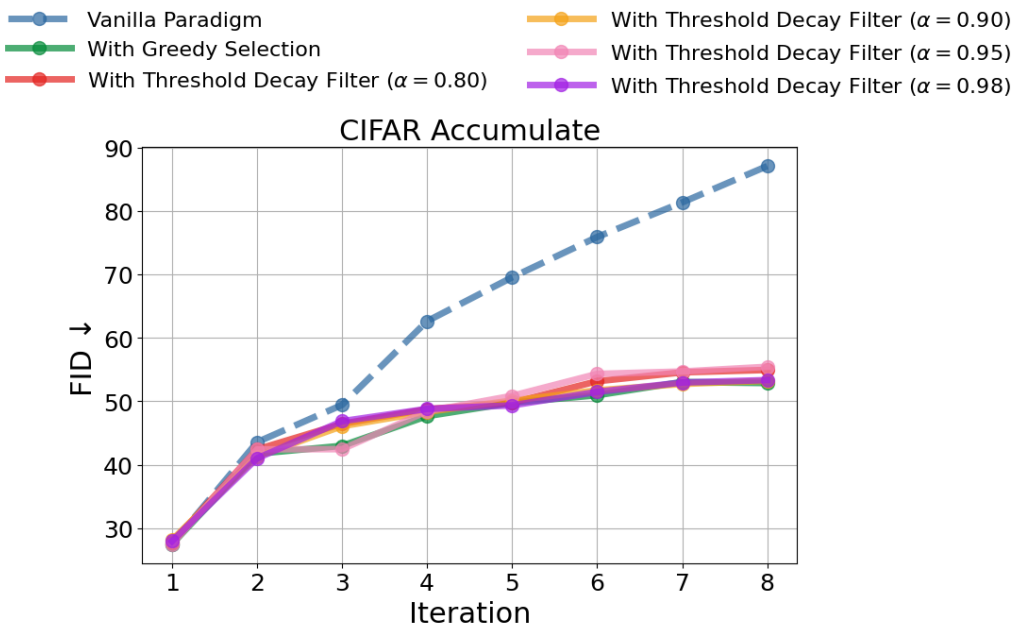


Figure 14: **FID score of the generated images over iterations.** We compare the results for different decay rates on CIFAR-10 dataset.

Figure 15: Generated images of models trained on **1024** CIFAR-10 images over iterations.

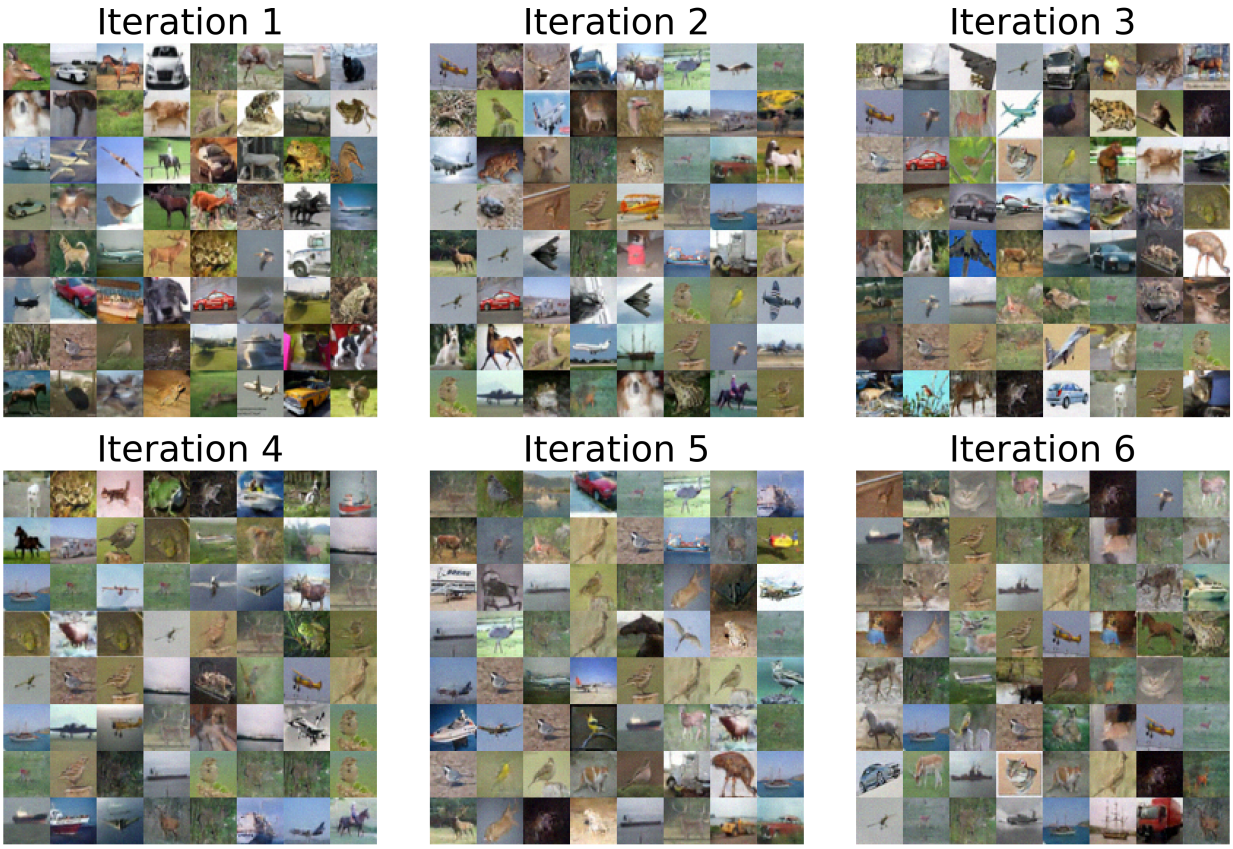


Figure 16: Generated images of models trained on **2048** CIFAR-10 images over iterations.

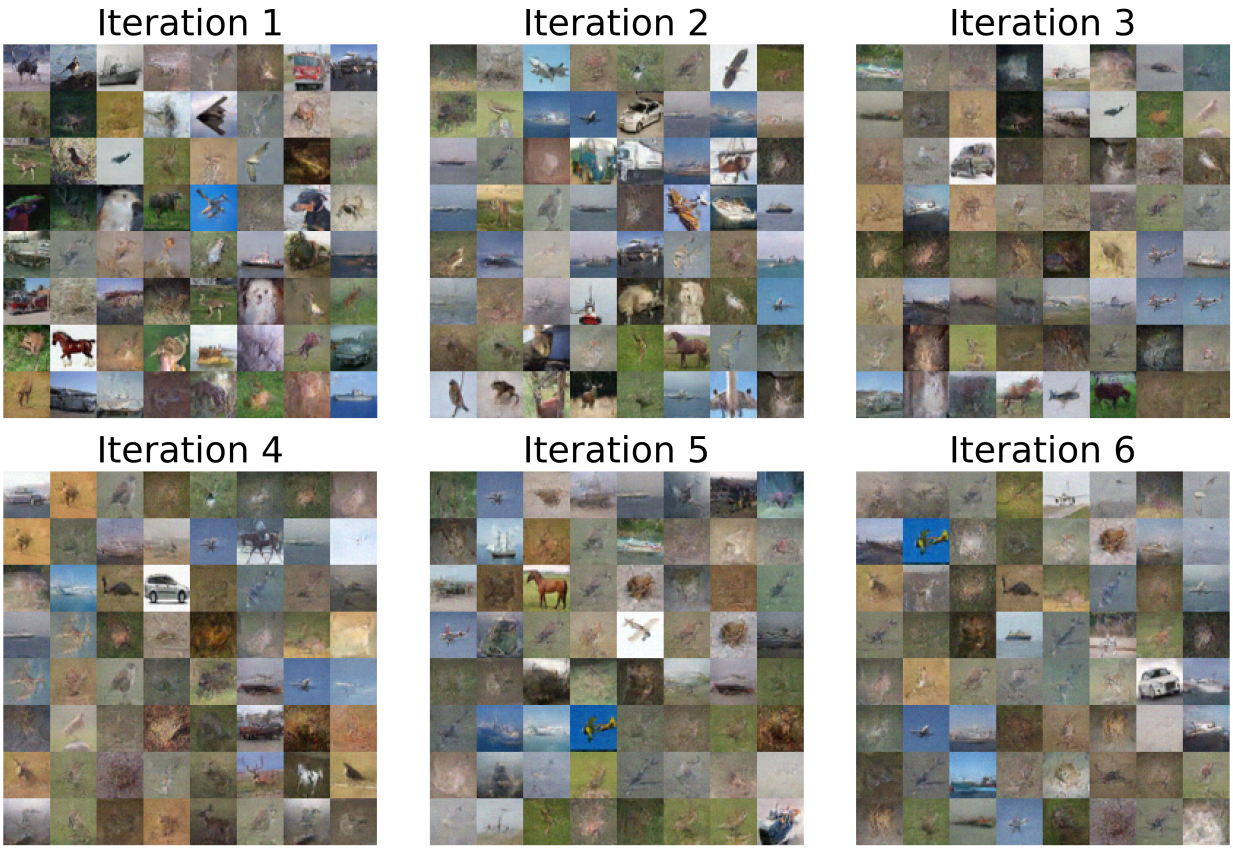


Figure 17: Generated images of models trained on **4096** CIFAR-10 images over iterations.

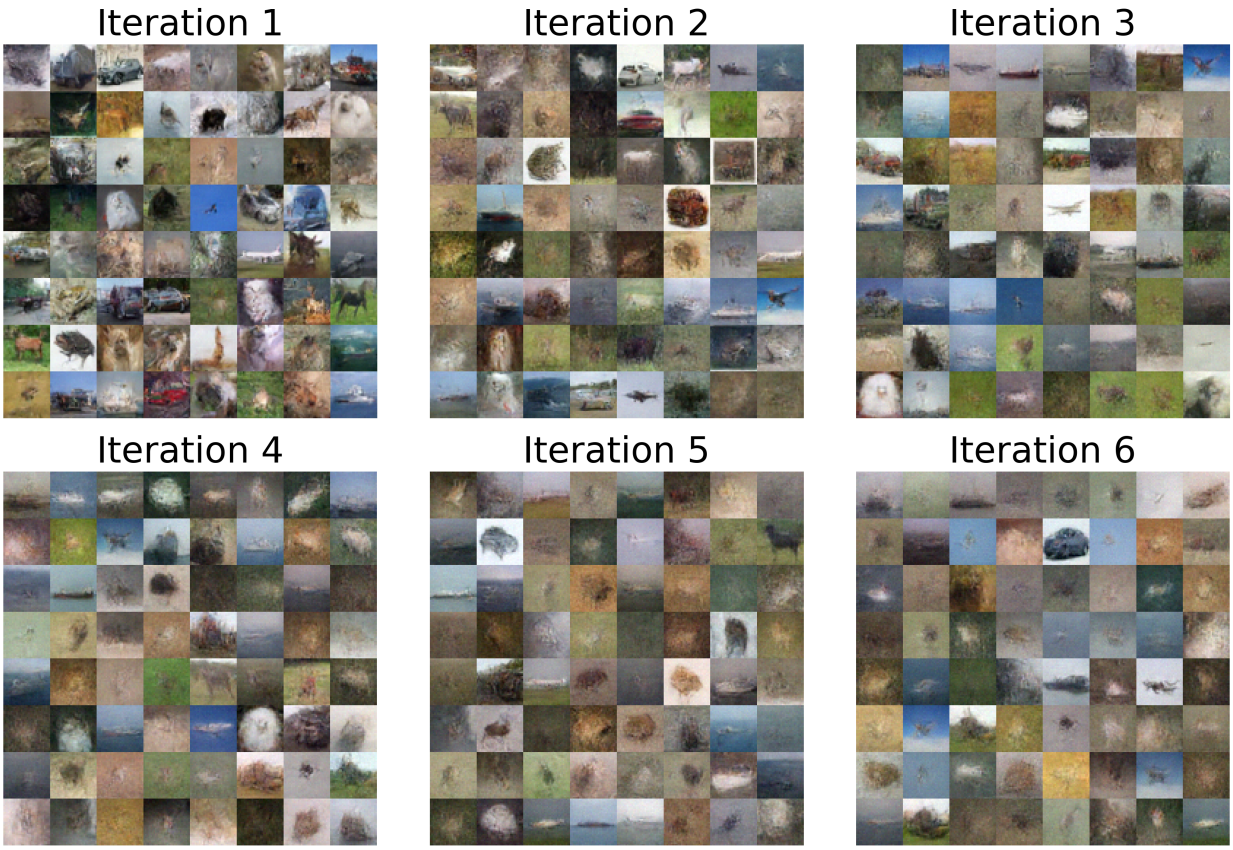


Figure 18: Generated images of models trained on **8192** CIFAR-10 images over iterations.

Figure 19: Generated images of models trained on **16384** CIFAR-10 images over iterations.

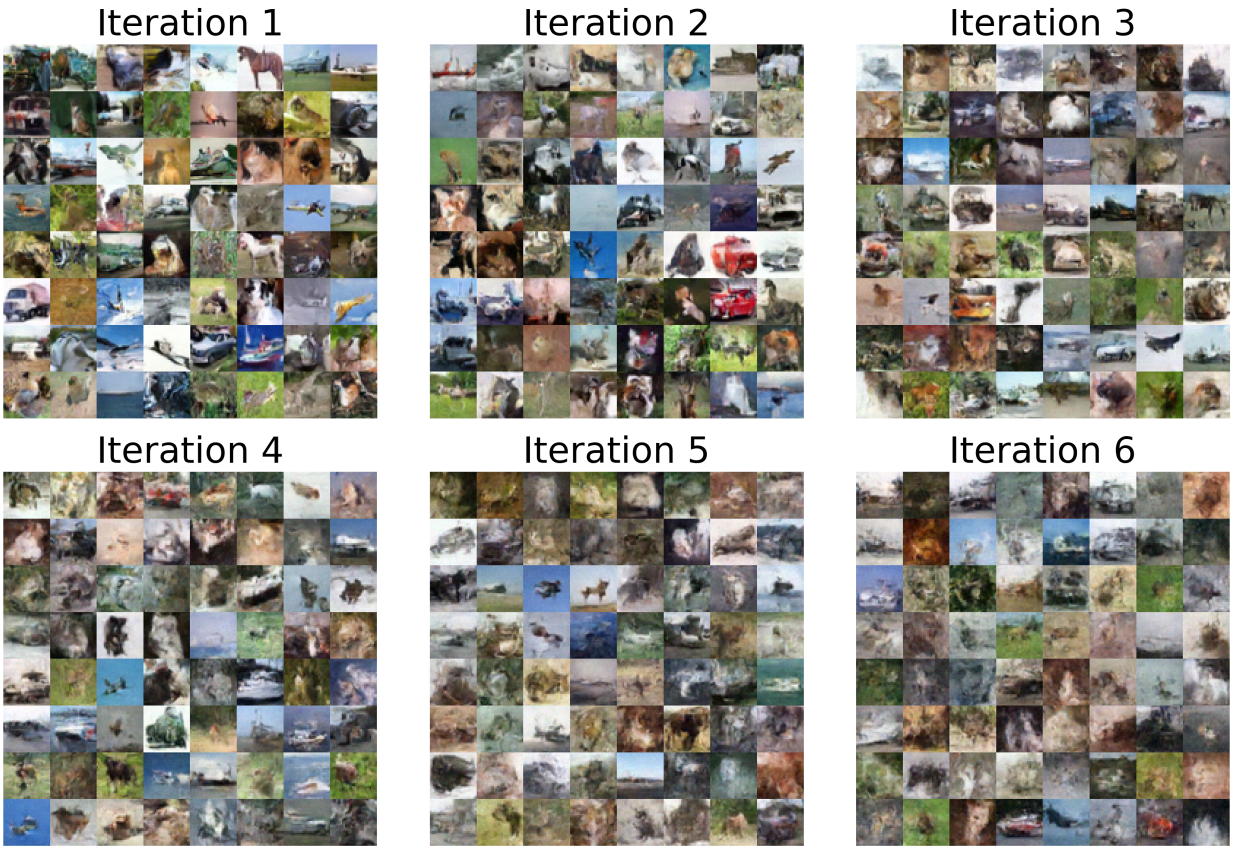


Figure 20: Generated images of models trained on **32768** CIFAR-10 images over iterations.