Exploring the Impact of Dataset Bias on Dataset Distillation

Yao Lu[1*] Jianyang Gu[2] Xuguang Chen[1] Saeed Vahidian[3] Qi Xuan[1†]
[1]Zhejiang University of Technology [2]Zhejiang University [3]Duke University

Abstract

Dataset Distillation (DD) is a promising technique to synthesize a smaller dataset that preserves essential information from the original dataset. This synthetic dataset can serve as a substitute for the original large-scale one, and help alleviate the training workload. However, current DD methods typically operate under the assumption that the dataset is unbiased, overlooking potential bias issues within the dataset itself. To fill in this blank, we systematically investigate the influence of dataset bias on DD. Given that there are no suitable biased datasets for DD, we first construct two biased datasets, CMNIST-DD and CCIFAR10-DD, to establish a foundation for subsequent analysis. Then we utilize existing DD methods to generate synthetic datasets on CMNIST-DD and CCIFAR10-DD, and evaluate their performance following the standard process. Experiments demonstrate that biases present in the original dataset significantly impact the performance of the synthetic dataset in most cases, which highlights the necessity of identifying and mitigating biases in the original datasets during DD. Finally, we reformulate DD within the context of a biased dataset. Our code along with biased datasets are available at https://github.com/yaolu-zjut/Biased-DD.

1. Introduction

Recently, Dataset Distillation (DD) has attracted widespread attention within the deep learning community due to its potential to alleviate data burden and enhance training efficiency. It was first introduced by Wang et al. [39], with the objective of condensing a large dataset into a small, synthetic one such that models trained on the latter yield comparable performance. After that, lots of subsequent study [3, 5, 8, 24, 25, 27, 32, 43, 47, 49] has proposed a series of methods to improve the performance of synthetic datasets, including gradient matching [19, 45, 47], trajectory matching [3, 5, 8, 9] and distribution matching approaches [32, 38, 46, 48]. Despite achieving significant improvements, existing DD methods usually operate under the presupposition that the dataset is unbiased, overlooking potential issues within the dataset itself. However, in reality, datasets can be fraught with various problems, such as bias [11, 18, 35], imbalance [6, 16], label noise [2, 34], and missing values [10, 12], which can significantly affect the reliability and effectiveness of machine learning models and algorithms trained on these datasets. So, what happens when DD encounters dataset issues? How do dataset issues affect DD? Yet, there hasn't been research (either empirical or theoretical) that can answer this question.

To fill in this blank, we aim to investigate the influence of dataset issues on DD. In this paper, we concentrate on dataset bias, which arises when unintended attributes (i.e., bias attributes) are highly correlated with the label attribute within the dataset. For example, many images labeled as "camel" may have a "desert" background, creating an unintentional correlation. In this way, models mistakenly associate "camel" with "desert" instead of learning the actual characteristics of a camel. In this case, the desert is a dataset bias.

First of all, we create two biased datasets for DD, named CMNIST-DD and CCFAR10-DD, following the instructions of Nam et al. [26]. Each dataset consists of 6 training sets with varying biased ratios (0%, 10%, 50%, 80%, 95% and 100%) and 1 unbiased testing set. We hope that these datasets can facilitate subsequent analysis on biased DD. Then we use several representative DD methods [45–47] to generate synthetic datasets on CMNIST-DD and CCIFAR10-DD and evaluate their performance with the default parameter setting in the original papers. Experimental results demonstrate that dataset bias does affect DD in most cases. Therefore, it is essential to consider potential biases in datasets during DD. In view of this, we further provide a mathematical definition of DD with biased datasets, which we termed "biased DD" below. Compared to vanilla DD [3, 24, 27, 47, 49], which aims to generate a small synthetic dataset that preserves as much information as possible from the original dataset, biased DD emphasizes unbiased attributes instead of the whole samples while minimizing the impact of biased attributes. We leave the specific implementation of biased DD to future work.

In summary, we emphasize our contributions as follows:

*yaolu.zjut@gmail.com
†Corresponding author, xuanqi@zjut.edu.cn.

- We propose a novel distillation scenario: distill valid information of large biased training sets into small, synthetic ones. To the best of our knowledge, We are among the few who consider dataset biases during DD.
- We create two biased datasets, named CMNIST-DD and CCFAR10-DD, to establish a foundation for subsequent analysis and the design of future debiased DD methods.
- Having obtained CMNIST-DD and CCFAR10-DD, we conduct comprehensive experiments on them and conclude that dataset biases can seriously affect the performance of DD in most cases, which urgently calls for bias mitigation strategies specifically tailored for DD. Besides, we redefine DD when distilling biased datasets and leave the specific implementation to future work.

2. Related Work

In this section, we briefly overview various dataset issues and existing work on DD.

Dataset Issues. Despite deep learning has achieved remarkable success in various fields [1, 29, 30], the datasets used to train these models contain many issues that cannot be ignored. One critical issue is dataset bias [11, 18, 35, 44], which arises when unintended attributes are highly correlated with the label. Such bias can lead to models that trained on these datasets producing inaccurate or unfair predictions. Another critical issue is data imbalance [20, 23, 28], where certain classes are overrepresented in the dataset, resulting in models skewed towards those majority classes and performing poorly on minority classes. Additionally, due to the expensiveness of the labeling process or difficulty of correctly classifying data (even for the experts), label noise [2, 34] becomes another common problem, which severely degrades the generalization performance of models. Missing values is also a common issue often attributed to human error, machine error, etc., and can cause performance degradation and data analysis problems.

Dataset Distillation, a method of compressing large datasets into smaller ones to improve training efficiency, was initially introduced by Wang et al. [39]. After that, many subsequent studies have introduced various matching losses to improve the performance of synthetic datasets. For example, DC [47], DSA [45] and IDC [19] are proposed to match gradients between synthetic and original samples. MTT [3], LCMat [33], FTD [8], TESLA [5] and DATM [15] introduce a trajectory matching paradigm to minimize the loss of training trajectories between synthetic and original datasets. Different from matching in parameter space, DM [46], CAFE [38], IDM [48] and DataDAM [32] use the feature space as the match proxy, and CLoM [25] utilizes pre-trained models to enhance the performance and cross-architecture generalization of synthetic datasets. Besides, DD has found extensive applications across various domains, including continual learning [14, 41], privacy protection [7, 36], federated learning [40, 42] and recommender systems [31, 37].

Despite existing studies have demonstrated the effectiveness of DD and its application across various fields, they all hinge on the assumption of an unbiased dataset. Our study is dedicated to exploring DD under dataset bias, a topic that stands orthogonal to, yet distinct from, existing study. To the best of our knowledge, prior to our work, only Cui et al. [4] consider dataset bias during DD. Specifically, they introduce a sample reweighting scheme that utilizes kernel density estimation to reduce bias in DD.

3. Preliminaries

In this section, we present the formulation of dataset bias (Sec. 3.1) and vanilla DD (Sec. 3.2).

3.1. Definition of Dataset Bias

Dataset bias is a dataset problem that occurs when unintended bias attributes hold a substantial correlation with the target attribute within the training dataset. To be specific, suppose x is a biased image sampled from a dataset with its corresponding label y, z is the attributes extracted from x. Among these attributes, z_g denotes the attribute that is essential for predicting a target label y, while z_b denotes the attribute that is less essential, but has a strong correlation with y. Since z_b is easier for the model to learn compared to z_g [26], the model becomes biased by overly exploiting z_b instead of z_g when trained on the biased dataset, failing to predict the samples which do not contain z_b.

For instance, many images labeled as "camel" may contain a "desert" background. This unintentional correlation will mislead models into associating "camel" with "desert" instead of learning the actual characteristics of a "camel". Samples that have a strong correlation (like "camel in the desert") are called **bias-aligned samples**, while samples that have a weak correlation (like "camel on the grass") are termed **bias-conflicting samples**. Finally, the biased rate of the dataset can be calculated by Equation (1), where N_{ba} and N_{bc} denote the number of bias-aligned samples and bias-conflicting samples respectively.

$$\textit{Biased Rate} = \frac{N_{ba}}{N_{bc} + N_{ba}} \tag{1}$$

3.2. Definition of Vanilla DD

Vanilla DD is built on the assumption of an unbiased dataset. Its goal is to generate a small synthetic dataset that retains as much information as possible from the original dataset.

Assume that we are given a large training set $\mathcal{T} = \{(x_i, y_i)\}|_{i=1}^{|\mathcal{T}|}$, the synthetic dataset $\mathcal{S} = \{(\hat{x}_i, y_i)\}|_{i=1}^{|\mathcal{S}|}$ ($|\mathcal{S}| \ll |\mathcal{T}|$), generated by vanilla DD, can be obtained

Figure 1. Visualizations of bias-conflicting samples and bias-aligned samples. Figure (a) and (f) visualize the bias-conflicting samples in CMNIST-DD and CCIFAR10-DD, respectively. Figure (b)-(e) and (g)-(j) visualize the bias-aligned samples with various severities in CMNIST-DD and CCIFAR10-DD, respectively. Severity increases from top to bottom. As for CCIFAR10-DD, we add 10 types of corruptions to 10 categories of CIFAR10. Specifically, "snow" for "airplane", "frost" for "automobile", "fog" for "bird", "brightness" for "cat", "contrast" for "deer", "spatter" for "dog", "elastic" for "frog", "JPEG" for "horse", "pixelate" for "ship" and "saturate" for "truck". Best viewed in color.

by solving the following minimization problem:

$$\min_{\mathcal{S}} \mathcal{D}(\mathcal{S}, \mathcal{T}), \tag{2}$$

where $\mathcal{D}$ is a task-specific matching loss.

4. Dataset Bias in Synthetic Datasets

4.1. Biased Dataset Preparation

Although biased datasets such as Colored MNIST and Corrupted CIFAR10 [26] already exist, these datasets exhibit high levels of bias (biased ratio: 95.0%, 98.0%, 99.0% and 99.5%), which is not conducive to comprehensively analyzing the impact of dataset bias on DD. To this end, we construct two biased datasets, CMNIST-DD and CCFAR10-DD, following the instructions of Nam et al. [26]. Specifically, as for CMNIST-DD, we select ten distinct colors and inject each color with random perturbation into the foreground of each digit of MNIST [22]. By adjusting the number of bias-aligned samples in the training set, we obtain six different datasets with the ratio of bias-aligned samples of 0%, 10%, 50%, 80%, 95% and 100%. As for CCFAR10-DD, we utilize a set of protocols [17] for corruption and inject

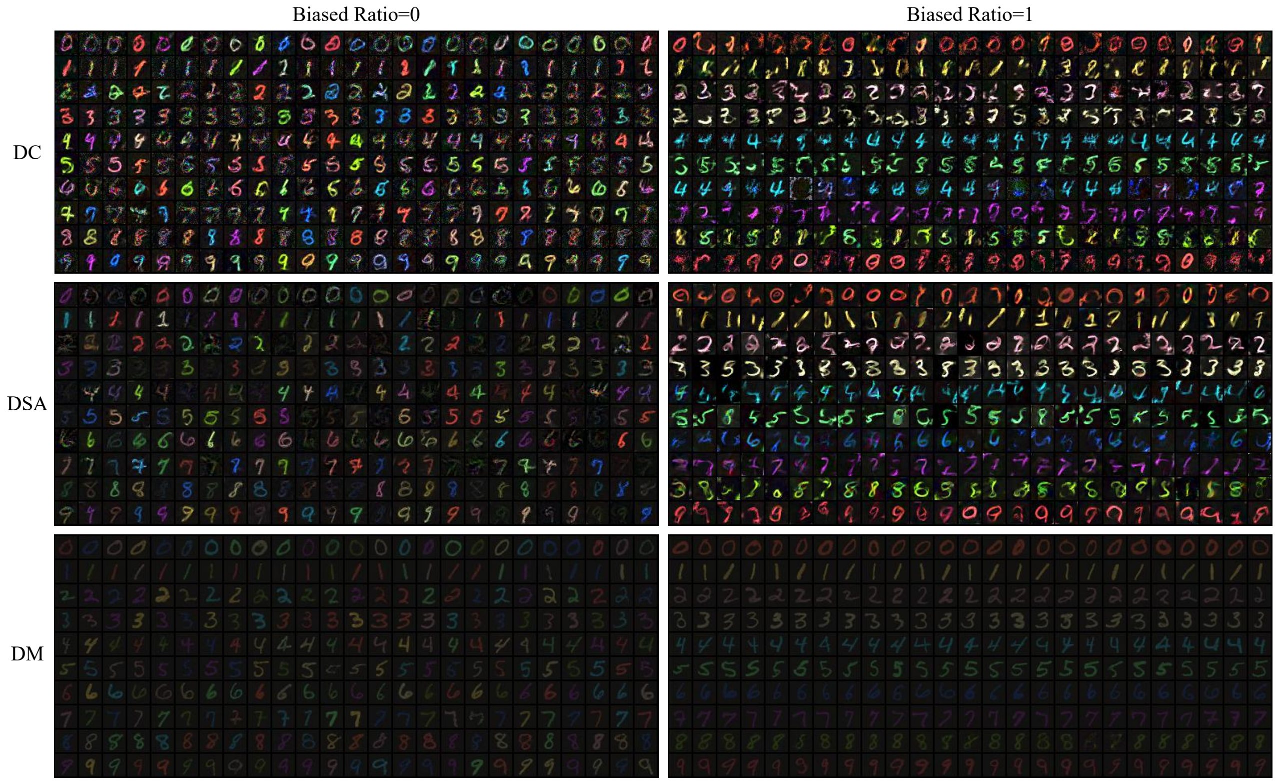

Figure 2. Visualizations of synthetic datasets generated by various DD methods on CMNIST-DD. All experiments are conducted at a severity level of 4.

them into CIFAR10 [21]. Specifically, "snow" for "airplane", "frost" for "automobile", "fog" for "bird", "brightness" for "cat", "contrast" for "deer", "spatter" for "dog", "elastic" for "frog", "JPEG" for "horse", "pixelate" for "ship" and "saturate" for "truck". CCFAR10-DD also has six different datasets with their correlation ratios as in CMNIST-DD. Finally, a parameter, *severity*, is introduced to regulate the intensity of disturbance on CMNIST-DD and CCFAR10-DD datasets. Figure 1 exhibits the bias-aligned samples under severity=1-4 for CMNIST-DD and CCFAR10-DD.

4.2. Experimental Setups

Having obtained CMNIST-DD and CCFAR10-DD, we next utilize them as the original dataset and perform DD on them. In this paper, we select three representative DD methods for experiments: gradient-matching based DC [47], DSA [45] and distribution-matching based DM [46].

Specifically, DC aligns the training gradients derived from synthetic samples with those obtained from original samples. Given a model with parameters θ, the optimization process can be expressed as:

$$\min_{\mathcal{S}} \sigma\left(\nabla_\theta \mathcal{L}(\theta;\mathcal{S}), \nabla_\theta \mathcal{L}(\theta;\mathcal{T})\right), \tag{3}$$

where $\mathcal{L}(\cdot;\cdot)$ denotes the training loss and $\sigma(\cdot;\cdot)$ represents the distance measure. On the basis of DC, DSA further applies data augmentation techniques to improve the performance of synthetic datasets:

$$\min_{\mathcal{S}} \sigma\left(\nabla_\theta \mathcal{L}(\mathcal{A}(\mathcal{S},\omega^{\mathcal{S}}),\theta), \nabla_\theta \mathcal{L}(\mathcal{A}(\mathcal{T},\omega^{\mathcal{T}}),\theta)\right), \tag{4}$$

where $\mathcal{A}$ is a family of image transformations such as cropping, color jittering and flipping that are parameterized with $\omega^{\mathcal{S}}$ and $\omega^{\mathcal{T}}$ for synthetic and real training sets respectively. DM aligns the feature distributions of synthetic and real training sets using maximum mean discrepancy [13] in sampled embedding spaces:

$$\min_{\mathcal{S}} \| \frac{1}{|\mathcal{T}|}\sum_{i=1}^{|\mathcal{T}|} f(\theta;\mathcal{A}(x_i,\omega^{\mathcal{T}})) - \frac{1}{|\mathcal{S}|}\sum_{i=1}^{|\mathcal{S}|} f(\theta;\mathcal{A}(s_i,\omega^{\mathcal{S}}))\|^2, \tag{5}$$

where $f(\cdot;\cdot)$ is the feature extraction function.

Implementation Details. In this paper, we use the default hyperparameter settings of DC, DSA and DM[1] to synthesize datasets and evaluate their performance. As for evaluating the original CMNIST-DD and CCIFAR10-DD, we set the batch size, weight decay, epoch and momentum to 256, 0.0005, 150 and 0.9, respectively. The optimizer is set as

[1] https://github.com/VICO-UoE/DatasetCondensation

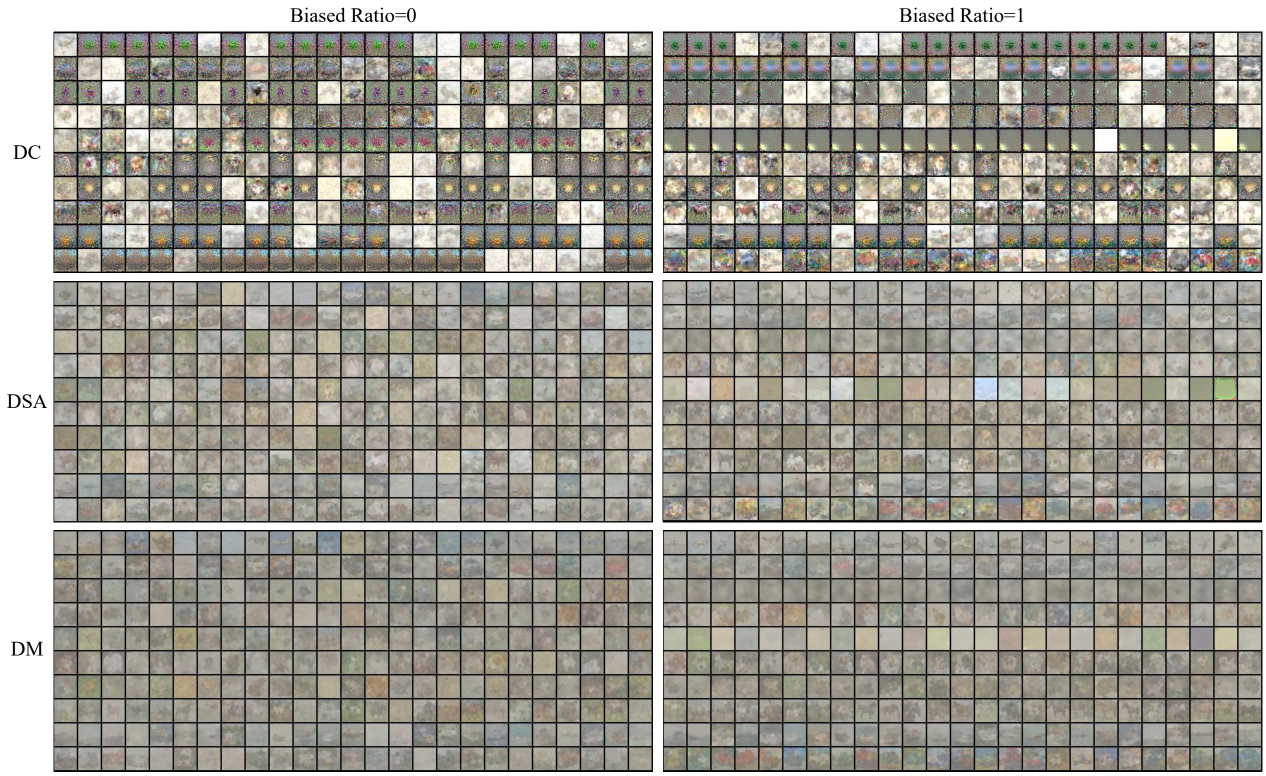

Figure 3. Visualizations of synthetic datasets generated by various DD methods on CCIFAR10-DD. All experiments are conducted at a severity level of 4.

SGD, with an initial learning rate of 0.01. The learning rate is decayed by a factor of 0.1 at epochs 50 and 100. Besides, we repeat each experiment 3 times and report the mean and standard deviation.

4.3. Experiments

We use the default hyperparameter settings of DC, DSA and DM to synthesize datasets (50 images per class) on CMNIST-DD and CCIFAR10-DD. Specifically, we use datasets with biased ratios of 0 and 1, at a severity level of 4 to conduct experiments. For clearer visualization, we select a subset of samples from the synthetic dataset and visualize them. Figure 2 and Figure 3 exhibit the visualizations of synthetic datasets generated by various DD methods on CMNIST-DD and CCIFAR10-DD, respectively. We find that when the biased ratio is 0, digits in the same class have completely different colors. However, when the biased ratio is 1, digits in the same class share the same color, which reveals that the color attribute has indeed been encoded into the synthetic datasets as a significant feature. As for CCIFAR10-DD, synthetic datasets generated from the biased dataset exhibit less diversity and richness compared to those derived from the unbiased dataset. We believe this phenomenon can be attributed to the biases in the original datasets that skew the distribution of features.

After that, we generate corresponding synthetic datasets using datasets with different biased ratios (0%, 10%, 50%, 80%, 95%, 100%) and evaluate their performance on an unbiased test set. Furthermore, we evaluate the performance of the model trained on the original dataset with various biased ratios as a control. As shown in Tab. 1, we observe that when the biased ratio is relatively low, the performance impact on synthetic CMNIST-DD is minimal. In other words, the performance of the synthetic dataset is relatively similar to that of the original CMNIST-DD. However, as the biased ratio increases ($\geq 50\%$), the performance disparity between the synthetic dataset and original CMNIST-DD is gradually increasing, which means DD is affected by dataset biases. When the biased ratio reaches 100%, the performance on the synthetic dataset experiences a dramatic decline, but the performance gap between the synthetic dataset and the original CMNIST-DD narrows significantly. As for CCIFAR10-DD, when the biased ratio is below 80%, the performance gap between the synthetic and original datasets is quite large, which means dataset biases do affect DD. However, when the biased ratio exceeds 95%, the performance of the synthetic dataset even higher than that of the original dataset. This phenomenon indicates that DD can retain more useful information of the original dataset under extreme bias rates (nearly 100%), offering a brand new perspective into the

Dataset	Method	Ratio of biaseds-aligned samples 0%	10%	50%	80%	95%	100%
CMNIST-DD	Full set	**99.49±0.02**	**99.49±0.02**	**99.29±0.01**	**98.50±0.05**	**95.22±0.11**	**8.89±0.50**
	DC	97.33±0.11	97.72±0.12	93.77±0.30	86.87±1.20	65.10±2.26	0.23±0.18
	DSA	98.08±0.10	98.18±0.06	97.26±0.04	95.34±0.10	84.29±0.90	1.44±0.14
	DM	97.42±0.02	97.32±0.04	94.03±0.32	74.15±0.09	12.92±0.53	6.45±0.91
CCIFAR10-DD	Full set	**73.77±0.35**	**72.86±0.12**	**67.13±0.36**	**55.02±0.18**	37.34±0.36	24.55±0.72
	DC	41.68±0.17	41.57±0.47	36.76±0.43	29.70±0.11	27.61±0.36	25.99±0.48
	DSA	49.52±0.29	48.72±0.50	42.86±0.35	37.45±0.44	34.24±0.61	33.31±0.46
	DM	52.29±0.51	52.54±0.25	47.47±0.73	42.13±0.21	**38.23±0.47**	**36.68±0.15**

Table 1. Performance of synthetic datasets (IPC=50) generated by different DD methods on CMNIST-DD and CCIFAR10-DD with varying ratios of bias-aligned samples (severity=4). Performance is evaluated on unbiased samples. "Full sets" means the model is trained on the original full dataset without distillation. **Bold entries** are best results.

Severity	1	2	3	4
CMNIST-DD BR=0	97.28±0.09	97.46±0.07	97.41±0.01	97.42±0.02
CMNIST-DD BR=100	8.65±0.24	7.78±1.02	6.60±0.36	6.45±0.91
CCIFAR10-DD BR=0	57.40±0.45	55.75±0.37	56.04±0.15	52.29±0.51
CCIFAR10-DD BR=100	49.92±0.10	50.06±0.44	45.10±0.13	36.68±0.15

Table 2. The effect of perturbation severity on the performance of synthetic datasets. BR denotes biased ratio. All experiments are conducted using DM, with 50 images per class.

design of dataset debiasing.

To delve into the impact of severity on DD, we conduct a series of experiments utilizing DM. By adjusting the severity (1-4) of the biased training set, we generate corresponding synthetic datasets and subsequently evaluate their performance on an unbiased test set. As illustrated in Tab. 2, the severity of disturbance also has a notable impact on the performance of synthetic datasets. Specifically, biased datasets are more susceptible to the increased disturbance severity, resulting in a more significant performance degradation than their unbiased counterparts.

In summary, although DD is less affected by dataset bias or even benefits from it at low and very high bias rates, but in most cases, dataset bias considerably impacts DD, which highlights the necessity of identifying and mitigating biases in the original datasets during DD.

5. Biased DD

In the previous section, we have demonstrated that dataset bias does affect DD and vanilla DD methods fail when faced with biased datasets in most cases, which indicates that the vanilla definition of DD is no longer suitable for biased datasets. To this end, we reformulate DD within the context of biased datasets, which we call biased DD, as follows:

Let $\mathcal{T}_b = \{(x_j, z_{g,j}, z_{b,j}, y_j)\}|_{j=1}^{|\mathcal{T}_b|}$ be the set of bias-aligned samples, where x_j and y_j are the j-th bias-aligned sample and its label, $z_{g,j}$ and $z_{b,j}$ denote its unbiased attribute and biased attribute. Similarly, the set of bias-conflicting samples can be formulated as $\mathcal{T}_g = \{(x_k, z_{g,k}, y_k)\}|_{k=1}^{|\mathcal{T}_g|}$, and $|\mathcal{T}_b| + |\mathcal{T}_g| = |\mathcal{T}|$. The objective of biased DD is to extract and retain the unbiased attributes z_g from both biased set $\mathcal{T}_b$ and unbiased set $\mathcal{T}_g$, while minimizing the impact of biased attributes z_b. Specifically, it can be formalized as the following optimization problem:

$$\min_{\mathcal{S}} \mathcal{D}(\mathcal{S}, \mathcal{A}_g) - \lambda \mathcal{D}(\mathcal{S}, \mathcal{A}_b), \tag{6}$$

where $\mathcal{A}_g$ is the composite of unbiased attributes present in both $\mathcal{T}_b$ and $\mathcal{T}_g$, $\mathcal{A}_b$ is the collection of biased attributes within $\mathcal{T}_b$. λ is a regularization balancing the contribution of unbiased and biased attributes in the optimization process. We will leave the specific implementation of biased DD to future work.

6. Conclusion

In this paper, we delve into DD when the dataset is endogenously biased. Specifically, we construct two biased datasets, namely CMNIST-DD and CCIFAR10-DD and conduct a series of experiments on them. Experimental results show that dataset biases indeed influence DD in most cases, highlighting the necessity of designing bias mitigation strategies specifically tailored for DD. Therefore, we propose a mathematical definition of biased DD and leave the specific implementation to future research.

Future Work. In this paper, we have demonstrated that dataset bias does affect DD. In the future, we will extend our experiments to larger datasets, more complex models and more advanced DD methods. Besides, it would be interesting to investigate why synthetic datasets outperform original datasets under extreme bias rates and what can be done with this phenomenon. Furthermore, how to eliminate or mitigate the impact of biased samples on synthetic datasets during the DD process is also a promising direction.

Acknowledgments

This work is supported in part by the Key R&D Program of Zhejiang under Grant 2022C01018 and by the National Natural Science Foundation of China under Grant U21B2001.

References

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. 2

[2] Görkem Algan and Ilkay Ulusoy. Image classification with deep learning in the presence of noisy labels: A survey. *Knowledge-Based Systems*, 215:106771, 2021. 1, 2

[3] George Cazenavette, Tongzhou Wang, Antonio Torralba, Alexei A Efros, and Jun-Yan Zhu. Dataset distillation by matching training trajectories. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4750–4759, 2022. 1, 2

[4] Justin Cui, Ruochen Wang, Yuanhao Xiong, and Cho-Jui Hsieh. Mitigating bias in dataset distillation. 2

[5] Justin Cui, Ruochen Wang, Si Si, and Cho-Jui Hsieh. Scaling up dataset distillation to imagenet-1k with constant memory. In *International Conference on Machine Learning*, pages 6565–6590. PMLR, 2023. 1, 2

[6] Debashree Devi, Saroj K Biswas, and Biswajit Purkayastha. A review on solution to class imbalance problem: Undersampling approaches. In *2020 international conference on computational performance evaluation (ComPE)*, pages 626–631. IEEE, 2020. 1

[7] Tian Dong, Bo Zhao, and Lingjuan Lyu. Privacy for free: How does dataset condensation help privacy? In *International Conference on Machine Learning*, pages 5378–5396. PMLR, 2022. 2

[8] Jiawei Du, Yidi Jiang, Vincent YF Tan, Joey Tianyi Zhou, and Haizhou Li. Minimizing the accumulated trajectory error to improve dataset distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3749–3758, 2023. 1, 2

[9] Jiawei Du, Qin Shi, and Joey Tianyi Zhou. Sequential subset matching for dataset distillation. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. 1

[10] Tlamelo Emmanuel, Thabiso Maupong, Dimane Mpoeleng, Thabo Semong, Banyatsang Mphago, and Oteng Tabona. A survey on missing data in machine learning. *Journal of Big data*, 8:1–37, 2021. 1

[11] Simone Fabbrizzi, Symeon Papadopoulos, Eirini Ntoutsi, and Ioannis Kompatsiaris. A survey on bias in visual datasets. *Computer Vision and Image Understanding*, 223:103552, 2022. 1, 2

[12] Raymond Feng, Flavio Calmon, and Hao Wang. Adapting fairness interventions to missing values. *Advances in Neural Information Processing Systems*, 36, 2024. 1

[13] Arthur Gretton, Karsten M Borgwardt, Malte J Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *The Journal of Machine Learning Research*, 13(1):723–773, 2012. 4

[14] Jianyang Gu, Kai Wang, Wei Jiang, and Yang You. Summarizing stream data for memory-restricted online continual learning. *arXiv preprint arXiv:2305.16645*, 2023. 2

[15] Ziyao Guo, Kai Wang, George Cazenavette, Hui Li, Kaipeng Zhang, and Yang You. Towards lossless dataset distillation via difficulty-aligned trajectory matching. *arXiv preprint arXiv:2310.05773*, 2023. 2

[16] Khan Md Hasib, Md Sadiq Iqbal, Faisal Muhammad Shah, Jubayer Al Mahmud, Mahmudul Hasan Popel, Md Imran Hossain Showrov, Shakil Ahmed, and Obaidur Rahman. A survey of methods for managing the classification and solution of data imbalance problem. *arXiv preprint arXiv:2012.11870*, 2020. 1

[17] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *arXiv preprint arXiv:1903.12261*, 2019. 3

[18] Aditya Khosla, Tinghui Zhou, Tomasz Malisiewicz, Alexei A Efros, and Antonio Torralba. Undoing the damage of dataset bias. In *Computer Vision–ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part I 12*, pages 158–171. Springer, 2012. 1, 2

[19] Jang-Hyun Kim, Jinuk Kim, Seong Joon Oh, Sangdoo Yun, Hwanjun Song, Joonhyun Jeong, Jung-Woo Ha, and Hyun Oh Song. Dataset condensation via efficient synthetic-data parameterization. In *International Conference on Machine Learning*, pages 11102–11118. PMLR, 2022. 1, 2

[20] Bartosz Krawczyk. Learning from imbalanced data: open challenges and future directions. *Progress in Artificial Intelligence*, 5(4):221–232, 2016. 2

[21] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 4

[22] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. 3

[23] Joffrey L Leevy, Taghi M Khoshgoftaar, Richard A Bauder, and Naeem Seliya. A survey on addressing high-class imbalance in big data. *Journal of Big Data*, 5(1):1–30, 2018. 2

[24] Yanqing Liu, Jianyang Gu, Kai Wang, Zheng Zhu, Wei Jiang, and Yang You. Dream: Efficient dataset distillation by representative matching. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 17314–17324, 2023. 1

[25] Yao Lu, Xuguang Chen, Yuchen Zhang, Jianyang Gu, Tianle Zhang, Yifan Zhang, Xiaoniu Yang, Qi Xuan, Kai Wang, and Yang You. Can pre-trained models assist in dataset distillation? *arXiv preprint arXiv:2310.03295*, 2023. 1, 2

[26] Junhyun Nam, Hyuntak Cha, Sungsoo Ahn, Jaeho Lee, and Jinwoo Shin. Learning from failure: De-biasing classifier from biased classifier. *Advances in Neural Information Processing Systems*, 33:20673–20684, 2020. 1, 2, 3

[27] Timothy Nguyen, Roman Novak, Lechao Xiao, and Jaehoon Lee. Dataset distillation with infinitely wide convolutional networks. *Advances in Neural Information Processing Systems*, 34:5186–5198, 2021. 1

[28] Minlong Peng, Qi Zhang, Xiaoyu Xing, Tao Gui, Xuanjing Huang, Yu-Gang Jiang, Keyu Ding, and Zhigang Chen. Trainable undersampling for class-imbalance learning. In *Proceedings of the AAAI conference on artificial intelligence*, pages 4707–4714, 2019. 2

[29] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 2

[30] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1(2): 3, 2022. 2

[31] Noveen Sachdeva, Mehak Dhaliwal, Carole-Jean Wu, and Julian McAuley. Infinite recommendation networks: A data-centric approach. *Advances in Neural Information Processing Systems*, 35:31292–31305, 2022. 2

[32] Ahmad Sajedi, Samir Khaki, Ehsan Amjadian, Lucy Z Liu, Yuri A Lawryshyn, and Konstantinos N Plataniotis. Datadam: Efficient dataset distillation with attention matching. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 17097–17107, 2023. 1, 2

[33] Seungjae Shin, Heesun Bae, Donghyeok Shin, Weonyoung Joo, and Il-Chul Moon. Loss-curvature matching for dataset selection and condensation. In *International Conference on Artificial Intelligence and Statistics*, pages 8606–8628. PMLR, 2023. 2

[34] Hwanjun Song, Minseok Kim, Dongmin Park, Yooju Shin, and Jae-Gil Lee. Learning from noisy labels with deep neural networks: A survey. *IEEE transactions on neural networks and learning systems*, 2022. 1, 2

[35] Antonio Torralba and Alexei A Efros. Unbiased look at dataset bias. In *CVPR*, pages 1521–1528. IEEE, 2011. 1, 2

[36] Margarita Vinaroz and Mi Jung Park. Differentially private kernel inducing points (dp-kip) for privacy-preserving data distillation. *arXiv preprint arXiv:2301.13389*, 2023. 2

[37] Cheng Wang, Jiacheng Sun, Zhenhua Dong, Ruixuan Li, and Rui Zhang. Gradient matching for categorical data distillation in ctr prediction. In *Proceedings of the 17th ACM Conference on Recommender Systems*, pages 161–170, 2023. 2

[38] Kai Wang, Bo Zhao, Xiangyu Peng, Zheng Zhu, Shuo Yang, Shuo Wang, Guan Huang, Hakan Bilen, Xinchao Wang, and Yang You. Cafe: Learning to condense dataset by aligning features. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12196–12205, 2022. 1, 2

[39] Tongzhou Wang, Jun-Yan Zhu, Antonio Torralba, and Alexei A Efros. Dataset distillation. *arXiv preprint arXiv:1811.10959*, 2018. 1, 2

[40] Yuanhao Xiong, Ruochen Wang, Minhao Cheng, Felix Yu, and Cho-Jui Hsieh. Feddm: Iterative distribution matching for communication-efficient federated learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16323–16332, 2023. 2

[41] Enneng Yang, Li Shen, Zhenyi Wang, Tongliang Liu, and Guibing Guo. An efficient dataset condensation plugin and its application to continual learning. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. 2

[42] Jie Zhang, Chen Chen, Bo Li, Lingjuan Lyu, Shuang Wu, Shouhong Ding, Chunhua Shen, and Chao Wu. Dense: Data-free one-shot federated learning. *Advances in Neural Information Processing Systems*, 35:21414–21428, 2022. 2

[43] Lei Zhang, Jie Zhang, Bowen Lei, Subhabrata Mukherjee, Xiang Pan, Bo Zhao, Caiwen Ding, Yao Li, and Dongkuan Xu. Accelerating dataset distillation via model augmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11950–11959, 2023. 1

[44] Quanshi Zhang, Wenguan Wang, and Song-Chun Zhu. Examining cnn representations with respect to dataset bias. In *Proceedings of the AAAI conference on artificial intelligence*, 2018. 2

[45] Bo Zhao and Hakan Bilen. Dataset condensation with differentiable siamese augmentation. In *International Conference on Machine Learning*, pages 12674–12685. PMLR, 2021. 1, 2, 4

[46] Bo Zhao and Hakan Bilen. Dataset condensation with distribution matching. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 6514–6523, 2023. 1, 2, 4

[47] Bo Zhao, Konda Reddy Mopuri, and Hakan Bilen. Dataset condensation with gradient matching. In *International Conference on Learning Representations*, 2020. 1, 2, 4

[48] Ganlong Zhao, Guanbin Li, Yipeng Qin, and Yizhou Yu. Improved distribution matching for dataset condensation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7856–7865, 2023. 1, 2

[49] Yongchao Zhou, Ehsan Nezhadarya, and Jimmy Ba. Dataset distillation using neural feature regression. *Advances in Neural Information Processing Systems*, 35:9813–9827, 2022. 1