How I Met Your Bias: Investigating Bias Amplification in Diffusion Models Supplementary Material

Nathan Roos[1] Ekaterina Iakovleva[1] Ani Gjergji[2] Vito Paolo Pastore[2,3*] Enzo Tartaglione[1*]
[1]LTCI, Télécom Paris, Institut Polytechnique de Paris, France
[2]MaLGa-DIBRIS, University of Genova, Italy
[3]AIGO, Istituto Italiano di Tecnologia, Italy

Digit class	RGB values of associated color	Color
0	$(255, 0, 0)$	red
1	$(0, 255, 0)$	green
2	$(0, 0, 255)$	blue
3	$(255, 255, 0)$	yellow
4	$(255, 0, 255)$	magenta
5	$(0, 255, 255)$	cyan
6	$(255, 128, 0)$	orange
7	$(255, 0, 128)$	rose
8	$(128, 0, 255)$	electric violet
9	$(128, 128, 128)$	grey

Table 1. RGB values of the colors associated with the digit classes in the synthetic dataset Biased MNIST.

Digit class	RGB values of left color	RGB values of right color
0	(250, 79, 42)	(4, 175, 212)
1	(252, 233, 89)	(2, 21, 165)
2	(171, 117, 147)	(83, 137, 107)
3	(199, 212, 153)	(55, 42, 101)
4	(22, 198, 250)	(232, 56, 4)
5	(81, 245, 113)	(173, 9, 141)
6	(6, 60, 193)	(248, 194, 61)
7	(141, 25, 194)	(113, 229, 60)
8	(52, 100, 4)	(202, 154, 250)
9	(212, 51, 68)	(42, 203, 186)

Table 2. RGB values of the colors associated with the digit classes in the synthetic dataset Multi-Color MNIST.

A. Datasets

We present here a more detailed description of the datasets employed for our experiments.

Biased MNIST. Biased MNIST is a synthetic variant of the MNIST handwritten digit dataset, originally introduced in [1], and widely used in the debiasing literature to evaluate the effectiveness of debiasing methods. The dataset is constructed by associating a specific background color to each of the ten digit classes (e.g., 0: red, 1: green, etc.; see Tab. 1). For a proportion ρ_{dataset} of the training samples, the background color matches the one assigned to the digit's class. For the remaining $1 - \rho_{\text{dataset}}$, a background color corresponding to a different class (selected uniformly at random) is applied. We report the RGB values of the colors associated with the digits in Biased MNIST in Tab. 1.

Multi-Color MNIST. Multi-Color MNIST is yet another synthetic variant of the MNIST handwritten dataset, originally introduced in [8] and widely used in the debiaising literature to evaluate the effectiveness of debiaising methods on multiple biases. The dataset is constructed by associating two specific background color to the ten digit classes (e.g., 0: left color is red, right color is light blue, etc.; see Tab. 2). For a proportion $\rho_{\text{left,dataset}}$ (respectively $\rho_{\text{right,dataset}}$) of the training samples, the left (respectively right) background color matches the one assigned to the digit's class. For the remaining $1 - \rho_{\text{left,dataset}}$ (resp. $1 - \rho_{\text{right,dataset}}$), a left (resp. right) background color corresponding to a different class (selected uniformly at random) is applied. $\rho_{\text{left,dataset}}$ and $\rho_{\text{right,dataset}}$ are fully tunable, thus allowing to experiment with different noise level combinations. We report the RGB values of the colors associated with the digits in Multi-Color MNIST in Tab. 2.

BFFHQ. Biased Flickr-Faces-HQ (BFFHQ) is a dataset of face images that builds upon FFHQ [7] by introducing demographic biases. The target of the generation is the age of the individual in the image, defined as $\mathcal{Y} = \{\text{“}\texttt{young}\text{”}, \text{“}\texttt{old}\text{”}\}$, while the bias attribute is gender, defined as $\mathcal{B} = \{\text{“}\texttt{female}\text{”}, \text{“}\texttt{male}\text{”}\}$. A proportion $\rho_{\text{dataset}} = 0.95$ of images labeled as “young” depict female subjects, while the same proportion of images labeled as “old” depict male subjects.

LAION-5B. LAION-5B [11] is a large-scale dataset consisting of image–captions pairs collected from the web. Due to its web-scraped nature, it inherently reflects a wide range of societal stereotypes [2, 3]. We focus on the gender as a bias attribute ($\mathcal{B} = \{\text{“}\texttt{female}\text{”}, \text{“}\texttt{male}\text{”}\}$), as it is

*Equal senior contribution

Parameter	Biased MNIST	BFFHQ
Resolutions	32-16-8	64-32-16-8
Residual blocks per resolution	2	4
Resolutions with attention	16	16
Channels per resolution	128-128-128	128-256-256-256
Attention heads	1	1
Attention blocks in encoder	4	4
Attention blocks in decoder	2	2
Nb trainable parameters	$8,797,955$	$61,804,931$

Table 3. Details of the architecture of the UNets used on the datasets Biased MNIST and BFFHQ.

widely studied in the literature [15]. The target concept for generation is the occupation "`lawyer`". Unlike Biased MNIST and BFFHQ, estimating the baseline value ρ_{dataset} in LAION-5B is non-trivial. As highlighted in [12], there exists a significant distributional shift between the captions used during model training and the prompts used at inference time for bias evaluation. Accurate estimation of ρ_{dataset} would require replicating the complex methodology proposed in [12], which we omit for the sake of simplicity. Consequently, in the context of LAION-5B, we will focus on whether ρ_{model} changes when the sampling hyperparameters do, and not on the bias amplification phenomenon.

B. Experimental details

U-Net architecture. In Tab. 3 we report the main details of the network architectures used in this paper. We implemented them in a newly written codebase based loosely on the implementation by Song *et al*. [1][13] and based on the model and sampler implementation of Karras *et al*. [6].[2]

Image generation. The sampler we use on models trained on Biased MNIST, Multi-Color MNIST and BFFHQ is the stochastic sampler of Karras *et al*. [6] (we also test VP-sampler [13] and DPM-Solver [9] in the supplementary material). We make it vary by changing the hyperparameters $\{n_{\text{steps}}, S_{\text{churn}}, S_{\text{tmin}}, S_{\text{tmax}}, w\}$ and by either using or not the second order correction. Regarding Stable Diffusion, we prompt it with "A portrait photo of a lawyer". We use the DDIM sampler, which allows us to control the number of sampling steps with n_{steps} and the stochasticity with the parameter η. $\eta = 0$ corresponds to deterministic sampling and $\eta = 1$ introduces as much variance in the process as in the ancestral sampling of DDPM [5].

Image selection. We keep all the images generated by models trained on Biased MNIST. We only keep an image generated by Stable Diffusion or the model trained on BFFHQ if the face of the individual in the image is clearly visible. We use the OpenCV 8-bit quantized version of the Single-Shot-Multibox face detector to detect the face. For the model trained on BFFHQ, we only keep the image if the face detected with the highest confidence has a confidence level above 0.999. For Stable Diffusion, the image is kept if: a single face is detected, the confidence is above 0.95, and the bounding box is at least 10 pixels away from all borders.

[1] https://github.com/yang-song/score_sde

[2] https://github.com/NVlabs/edm

n_{steps}	$\eta = 0$	$\eta = 0.3$	$\eta = 0.7$
6	5990	5908	5691
10	7127	7178	7031
15	6427	6412	4603
20	6045	5863	3212
25	5422	5344	5387
50	2697	2653	2651
100	1999	1950	1935
150	1328	1300	1292
200	988	940	973
250	673	650	667

Table 4. Number of samples generated by Stable Diffusion used to compute ρ_{model} for every n_{steps} and every η.

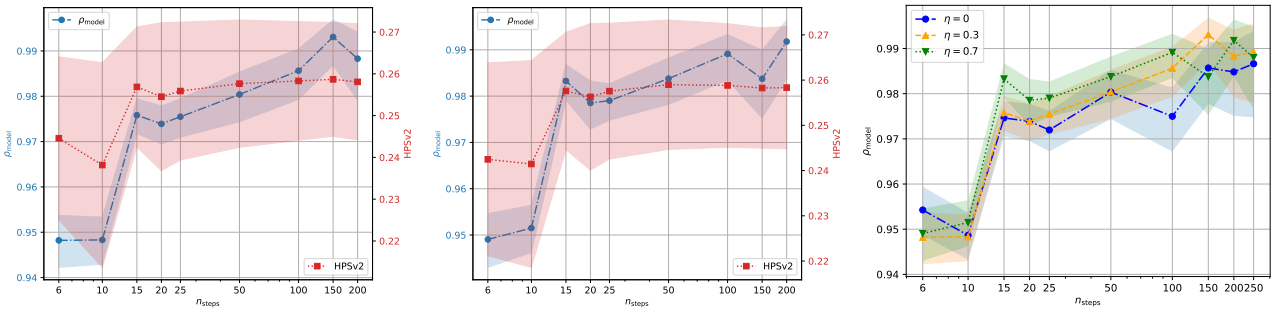
Image quality assessment. We evaluate the quality of the generated images for Stable Diffusion using the Human Preference Score v2 (HPSv2) [14], instead of the FID metric [4] used for Biased MNIST and BFFHQ. HPSv2 presents two key advantages over the FID: it eliminates the need for training samples with captions resembling the prompt and exhibits a higher correlation with human preferences. HPSv2 is trained by finetuning CLIP [10] on version 2 of the Human Preference Dataset [14].

Stable Diffusion image count. In Tab. 4, we report the number of images used to compute every point of Fig. 6a, Fig. 9a, Fig. 9b, and Fig. 9c. They vary for two reasons. First, we did not generate the same number of samples for every n_{steps} because the computational cost is linear in n_{steps}, and therefore we could generate more images for lower n_{steps}. Then, among the generated images, we had to remove those where the face was not clearly visible, following the process described in Sec. 4.1, which caused the number of remaining images to vary between the different η.

More details on results in the main paper. Here we summarize the specific configurations chosen to display the results in the main paper:

- Fig. 4: $\{n_{\text{steps}}{=}25, S_{\text{churn}}{=}80, S_{\text{tmin}}{=}0.01, S_{\text{tmax}}{=}80\}$. Each point is obtained by using the estimator in Eq. (8) on 4000 generated images. The unconditional score necessary for CFG is computed by averaging the conditional scores (as in Eq. (6)).
- Fig. 6a: $\{S_{\text{churn}}{=}80, S_{\text{tmax}}{=}80\}$.
- Fig. 6b: $\{S_{\text{churn}}{=}80, S_{\text{tmin}}{=}0.1\}$.
- Fig. 7: $\{S_{\text{tmin}}{=}0.1, S_{\text{tmax}}{=}80\}$.

Please note that the same effects described are observable for a broad set of hyperparameters choice - they are here chosen for visualization purposes only.



(a) ρ_{model} and HPSv2 vs n_{steps} with η=0.3. (b) ρ_{model} and HPSv2 vs n_{steps} with η=0.7. (c) ρ_{model} vs n_{steps}.

Figure 9. Stable Diffusion v2.1 using DDIM sampler with $\eta \in \{0, 0.3, 0.7\}$ (deterministic sampling). See Tab. 4 for the number of samples used to compute each point. Note that the x-axis is on a logarithmic scale.

C. More experimental results

Dataset with multiple biases. Our previous experiments focused on datasets with a single bias attribute. We now verify our claims on Multi-Color MNIST [8], a dataset with multiple known and controlled biases. In Multi-Color MNIST, the black background of each digit is split in two and filled with one color on the left and another color on the right. Similarly to Biased MNIST, the correlation between the colors and the digit is controlled by $\rho_{\text{left,dataset}}$ and $\rho_{\text{right,dataset}}$. Although the data set is quite simple, it is still relevant because the bias levels are fully tunable and we can assess ρ_{model} (left and right) with perfect accuracy. We present two combinations of the level of bias: one with medium bias $\{\rho_{\text{left,dataset}} = 0.9, \rho_{\text{right,dataset}} = 0.7\}$ and one with high bias $\{\rho_{\text{left,dataset}} = 0.99, \rho_{\text{right,dataset}} = 0.9\}$. In Fig. 10 and Fig. 11 we present the results for medium bias and in Fig. 12 and Fig. 13 the results for high bias. Our previous claims hold for $\rho_{\text{left,model}}$ and $\rho_{\text{right,model}}$ in both settings : n_{steps} and S_{churn} are positively correlated with $\rho_{\text{left,model}}$ and $\rho_{\text{right,model}}$. Moreover, for n_{steps} to have an effect on the level of bias, a minimal amount of noise in the sampling ($S_{\text{churn}} \approx 10$) is required. Overall, the conclusion that we have drawn from a single bias attribute remains valid for multiple biases.

Additional samplers for continuous framework. In addition to Karras' deterministic and stochastic samplers, we test two other samplers within the continuous framework. Specifically, we test the deterministic and stochastic VP-SDE sampler from Song *et al*. [13] and the deterministic DPM-Solver-1 [9].

The VP-SDE is a diffusion SDE introduced by Song *et al*. [13] in their seminal paper. It has a different noise schedule and scaling from the SDE used in EDM [6], resulting in a significantly different sampling trajectory. We refer to as the VP sampler the integration of the VP-SDE with the EDM scheme. The VP-SDE sampler has previously been studied and implemented by Karras *et al*. [6]. We measure ρ_{model} as incurred by the VP sampler on a model trained on 10-classes Biased MNIST using the same model and the same setup as in the experiment on the EDM sampler with results presented in Figure 8. Thus, we vary both S_{churn} and n_{steps}. The results with the VP sampler in Fig. 15 are extremely similar to those obtained with the EDM sampler, namely with a positive correlation between S_{churn} and ρ_{model} and between n_{steps} and ρ_{model}, as well as the fact that the values of ρ_{model} obtained with different n_{steps} for S_{churn}=0 are extremely close. This experiment shows that the results of our experiments on a given model carry over to a different sampler with different sampling trajectories, suggesting that the observed effects are a general phenomenon rather than specifics of a particular sampler.

DPM-Solver is a deterministic method to efficiently sample from the diffusion models leveraging the semi-linear structure of the probability flow ODE. In Fig. 14a and Fig. 14b, we observe that the ρ_{model} measured on a model trained on 2-classes Biased MNIST with DPM-Sampler-1 is independent with the number of sampling steps and positively correlated with the guidance scale. These findings are not new, but they corroborate the observation that in small models the number of sampling steps has an effect on the level of bias only when $S_{\text{churn}} > 0$, i.e., when the sampling process is stochastic rather than deterministic.

Effect of conditioning strength. We observe in Fig. 16 that with a high guidance scale, the biases of the classes from 0 to 5 are well represented and amplified as expected, but the biases from classes 6 to 9 are under-represented, or even not represented at all in the case of 9. It turns out that the RGB values (see Tab. 1) of the colors of classes 0 to 5 are in the corners of the cube $[0; 255]^3$ in the RGB space, whereas those of classes 6 to 8 are in the middle of the edges, and that the RGB value of the color correlated with class 9 is in the center of the cube. Keeping in mind that the effect of CFG is to add guidance (the second term in Eq. (5)) that "pushes" the samples away from the mean (unconditional) data distribution and towards the class dis-

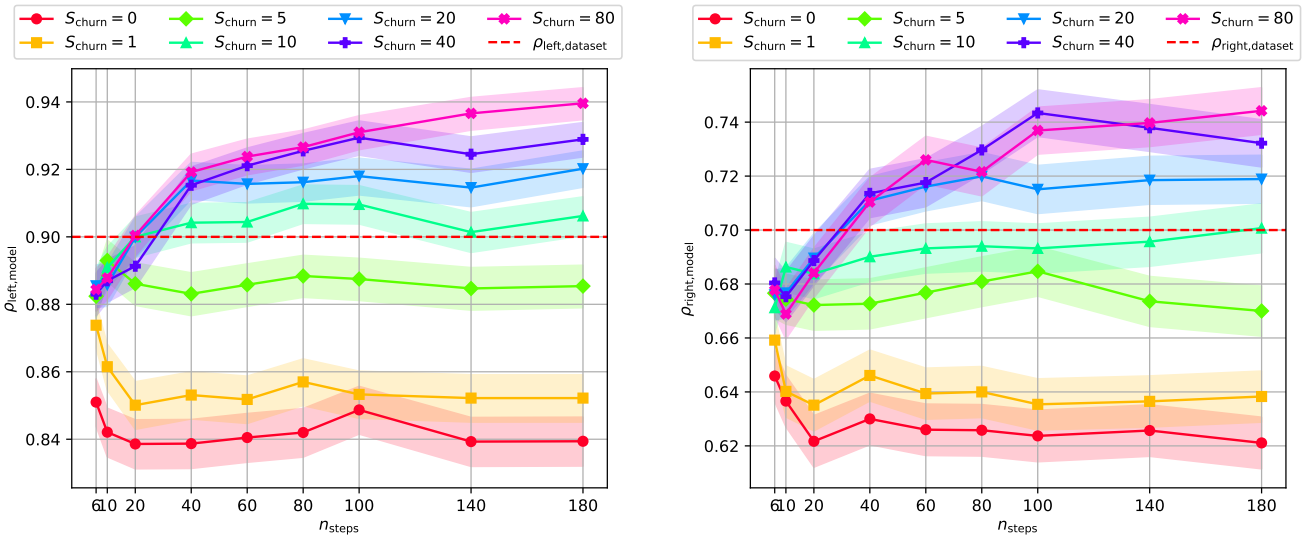


Figure 10. $\rho_{\text{left,model}}$ and $\rho_{\text{right,model}}$ vs n_{steps} for various S_{churn} for a model trained on Multi-Color MNIST ($\rho_{\text{left,dataset}}$=0.9, $\rho_{\text{right,dataset}}$=0.7) using EDM sampler (same experiment as Fig. 11).

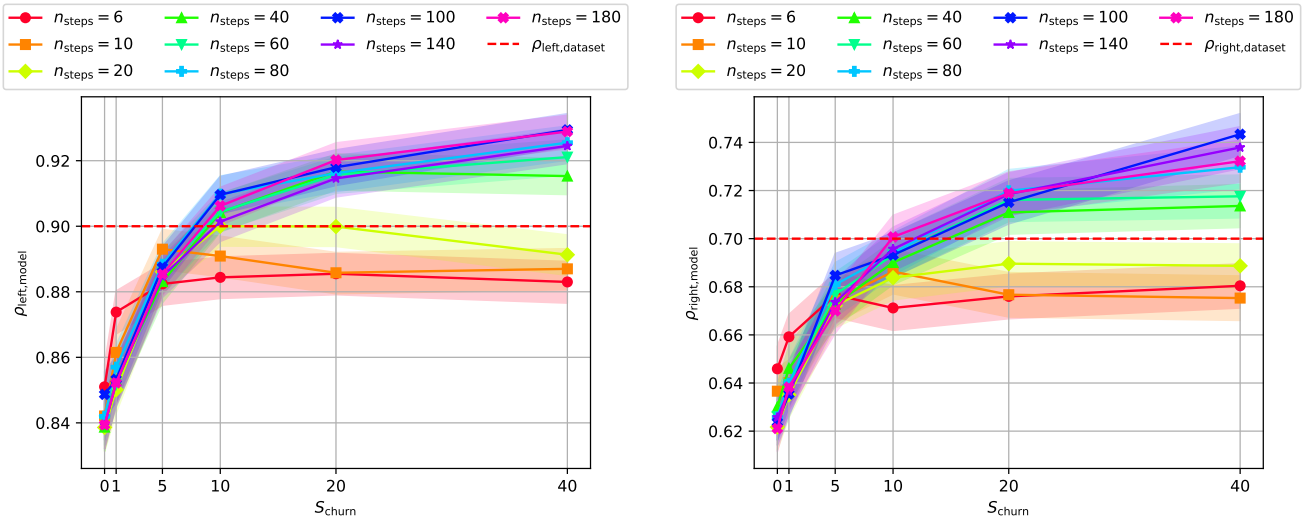


Figure 11. $\rho_{\text{left,model}}$ and $\rho_{\text{right,model}}$ vs S_{churn} for various n_{steps} for a model trained on Multi-Color MNIST ($\rho_{\text{left,dataset}}$=0.9, $\rho_{\text{right,dataset}}$=0.7) using EDM sampler (same experiment as Fig. 10).

tribution, we interpret this result as the CFG pushing the colors of the samples away from the mean color distribution (which is gray, at the center of the RGB cube), and towards the corners. Thus, the nature of the bias in 10-class Biased MNIST makes it unfit to properly study the effect of w, so we resort to 2-classes Biased MNIST, where the colors of both classes (red and green) play a symmetrical role.

Effect of the number of integration steps. We vary here the number of sampling steps n_{steps} for models trained on Biased MNIST and BFFHQ and for Stable Diffusion. In Fig. 6b and Fig. 18a we observe that in the 10-classes Biased MNIST ρ_{model} increases as n_{steps} increases (and likewise in Fig. 18a). More specifically, we can observe a *reduction* of bias at low n_{steps}, followed by an *amplification* of bias at high n_{steps}. The same is also evident in BFFHQ (Fig. 19a) and in Stable Diffusion (Fig. 6a). Furthermore, in Fig. 18a we remark that the range of values that ρ_{model} can take when we only vary n_{steps} is considerable: from 0.46 to 0.74. Therefore, the output distribution changes significantly, at least with respect to the bias, even though the quality of the generated digits does not change much as we see in Fig. 17. Fig. 19a shows the same trend on BFFHQ as

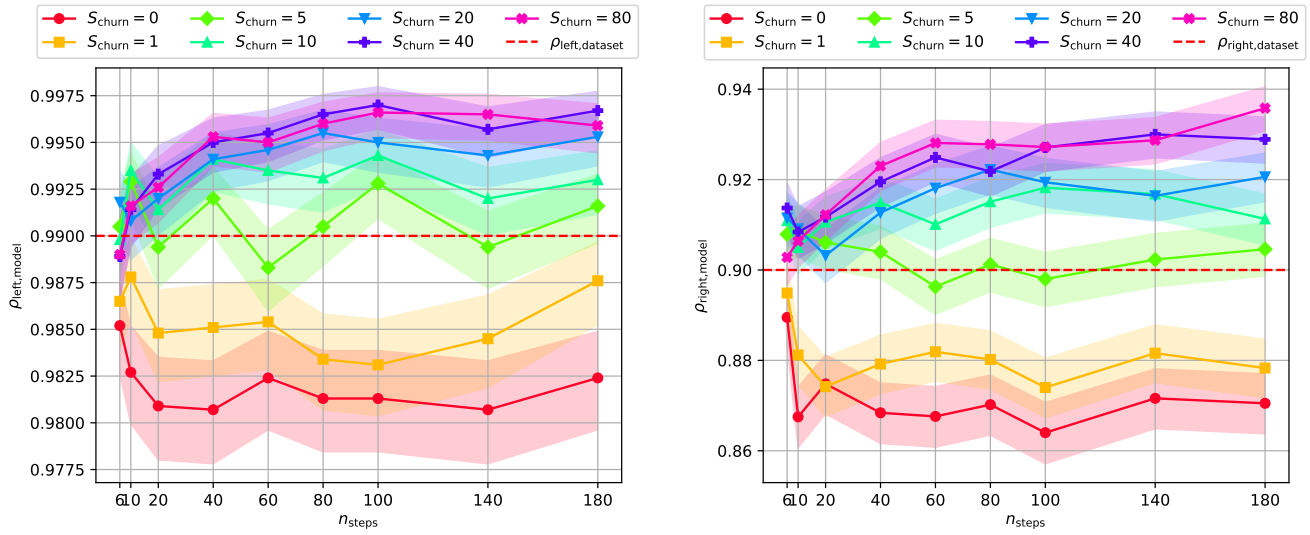


Figure 12. $\rho_{\text{left,model}}$ and $\rho_{\text{right,model}}$ vs n_{steps} for various S_{churn} for a model trained on Multi-Color MNIST ($\rho_{\text{left,dataset}}$=0.99, $\rho_{\text{right,dataset}}$=0.9) using EDM sampler (same experiment as Fig. 13).

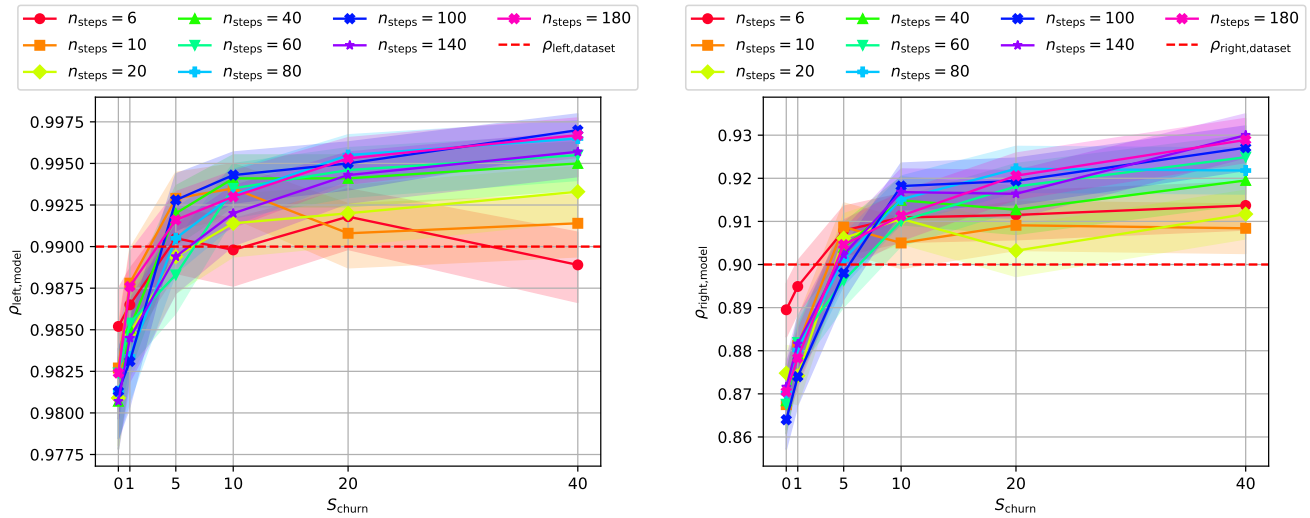


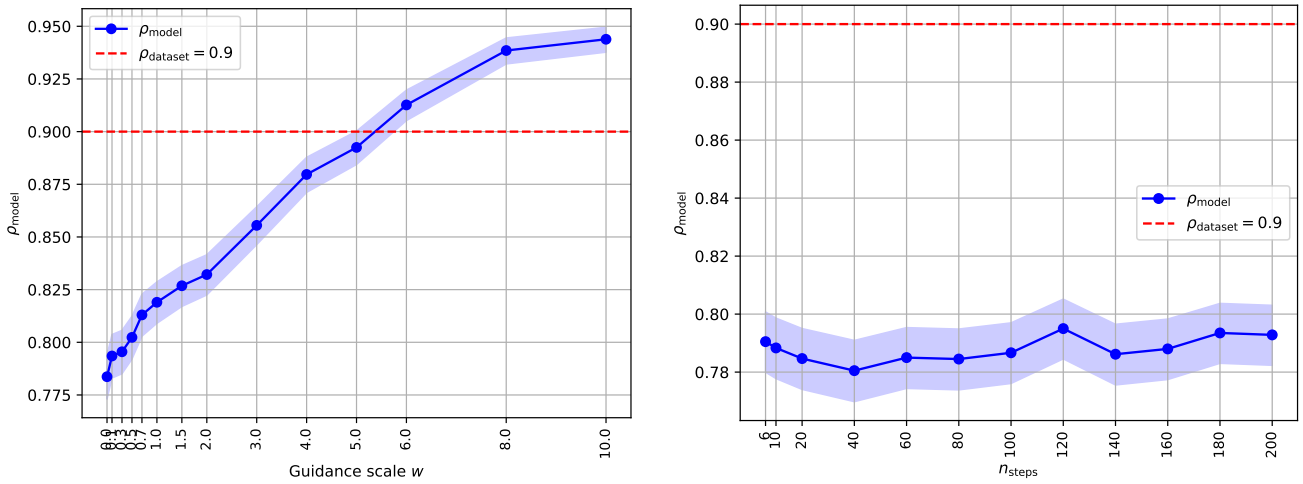
Figure 13. $\rho_{\text{left,model}}$ and $\rho_{\text{right,model}}$ vs S_{churn} for various n_{steps} for a model trained on Multi-Color MNIST ($\rho_{\text{left,dataset}}$=0.99, $\rho_{\text{right,dataset}}$=0.9) using EDM sampler (same experiment as Fig. 12).

previously observed on Biased MNIST: at low n_{steps} there is reduction of bias and at high n_{steps} the bias is amplified. Similarly, Fig. 6a shows the same increasing trend of ρ_{model} in Stable Diffusion. The trend remains when we use other values of η (see Fig. 9c). Since we do not have ρ_{dataset} as a baseline, we cannot conclude whether there is bias amplification, but we can at least observe that ρ_{model} varies significantly with n_{steps} (from 0.949 to 0.987), which is a previously unexplored phenomenon. In Fig. 6a we observe that although the HPSv2 score initially increases with n_{steps}, it plateaus quickly, while ρ_{model} continues to increase.

Time window with fresh noise. The two time windows in which the injected noise does not significantly change the generated distribution are $[40, 80]$ and $[0, 5]$:

- at the highest noise levels comprising $[40, 80]$, the bias (background color) may not yet be decided, hence the eventual variations due to stochasticity do not impact the generated color,
- at the lowest noise levels comprising $[0, 5]$, the characteristic features of the image (its digit and its color) have already appeared, and it remains only to refine the details.

Looking at the history of the denoised images in Fig. 20, we



(a) ρ_{model} vs w for a model trained on 2-classes Biased MNIST (ρ_{dataset}=0.9)) using DPM-Sovler-1 with hyperparameter $\{n_{\text{steps}} = 10\}$.

(b) ρ_{model} vs number of times teps on 2-classes Biased MNIST (ρ_{dataset}=0.9) using DPM-Sovler-1 with hyperparameter $\{w = 0\}$.

Figure 14. DPM-Solver-1 results on 2-classes Biased MNIST.

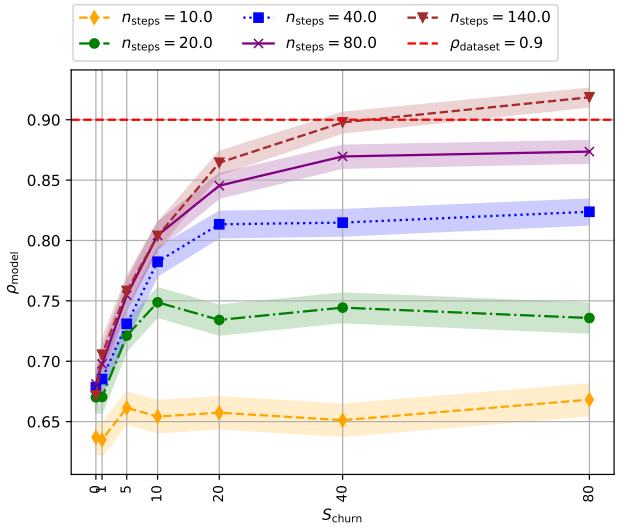


Figure 15. ρ_{model} vs S_{churn} for various n_{steps} for a model trained on 10-classes Biased MNIST (ρ_{dataset}=0.9) using VP sampler.

see that it is indeed possible to guess the end color starting from the time step $t \approx 15$. However, this might be an effect specific to Biased MNIST, as we did not replicate the experiment with other datasets and models.

The bias appears early. Overall, our interpretation of how the generation process produces biased images is that the bias already appears in early stages. As such, introducing noise into the sampling process during these stages can help reduce the upsurge of bias in generated images. We present, in Fig. 20, a qualitative visualization of how the denoising process produces samples in Biased MNIST (since the bias is the background color, it is easy to visually inspect). Indeed, the background color is the first generated feature, which supports our hypothesis. This could inspire more research in the field, notably conditioning generation of non-biased attributes in the early stages.

Robustness of the bias oracle. Since we use an oracle to obtain the gender and age labels of generated images on BFFHQ, we verify that this oracle is reliable by evaluating its accuracy on the train set. We also estimate the robustness of our oracle by classifying the noisy versions of the training data. To inject varying levels of noise, we add Gaussian noise with five different variances in the latent space of VQ-VAE in Stable Diffusion. The results for the full train set are shown in Fig. 21a and the group-wise accuracies are shown in Fig. 21b, where each group corresponds to a pair of (gender, age) labels. We can see that the gender prediction remains robust to an arbitrary level of noise added in the latent space. As for age prediction, the overall performance remains relatively stable, however, we do observe a temporary drop in acuracy by $\sim 3\%$ for the medium level of noise. We speculate that the local features in this case might be affected too strongly, just enough to resemble the fine lines on the face, but not enough to affect the background. To support our assumptions, we provide the noised images for all levels of noise considered in Fig. 22. As a result, the oracle makes more mistakes in the younger groups. If we continue to add the noise, the background also becomes perturbed because the injected noise is too large, and the local features are better preserved. We provide examples of the images generated with CDPM in Fig. 23 for reference.

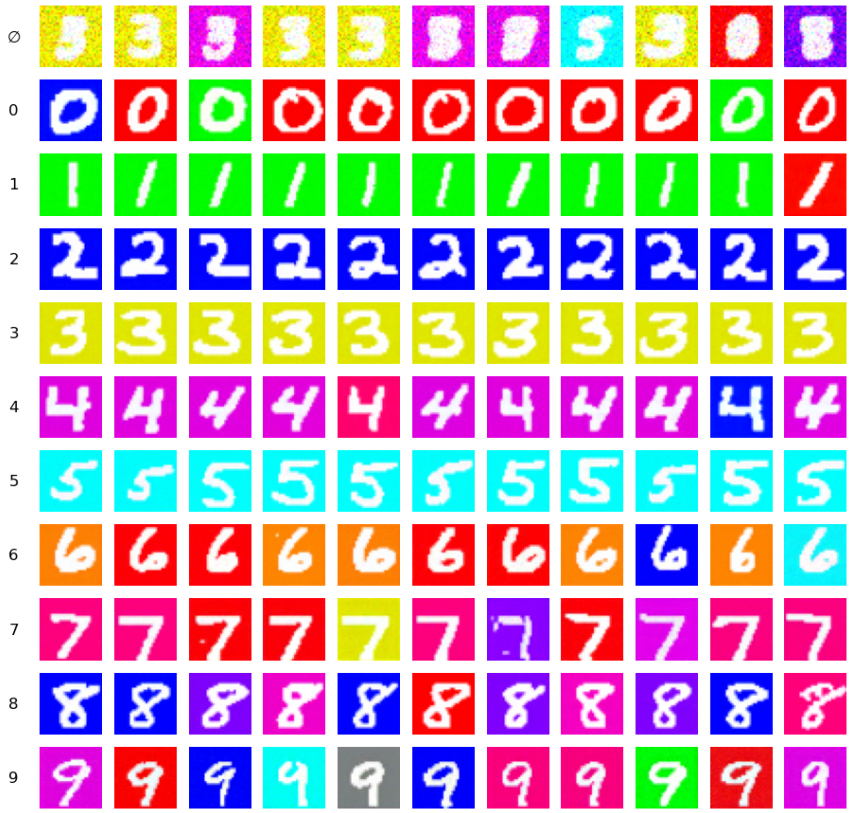
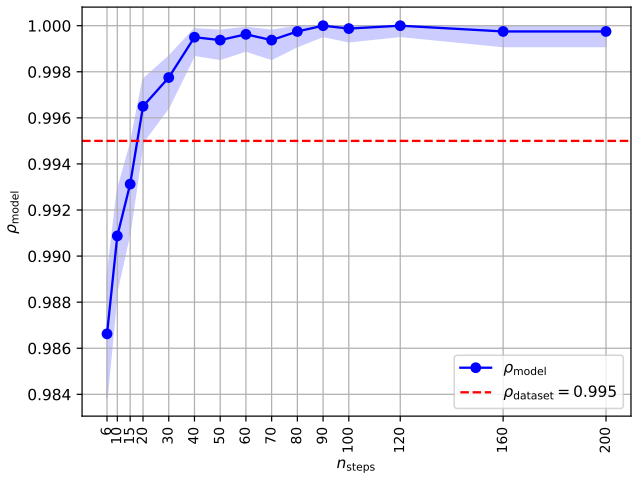
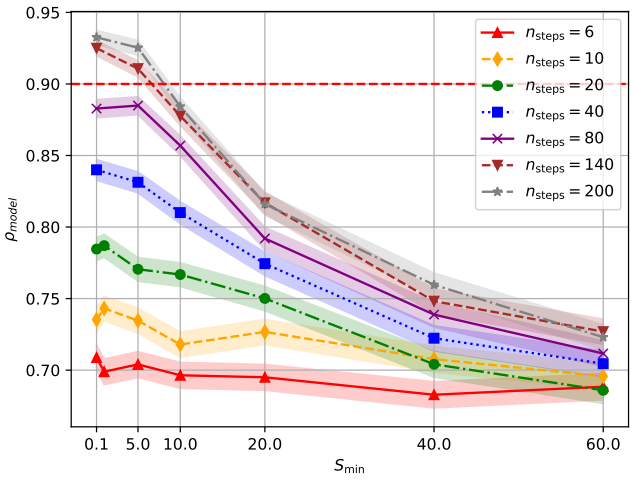


Figure 16. Samples generated by model trained on 10-classes Biased MNIST (ρ_{dataset}=0.9) using Karras deterministic sampler (S_{churn}=0, n_{steps}=12) with guidance scale w=12. Each row is sampled with a different conditioning: the first row are unconditional results, the others are obtained by conditioning on the class written at the beginning of the row. The unconditional score prediction is obtained by averaging the score prediction over all classes (see Eq. (6)).

Figure 17. Samples obtained with a model trained on Biased MNIST (ρ_{dataset}=0.7) using Karras stochastic sampler with hyperparameters $\{S_{\text{churn}}$=60, S_{tmin}=0.01, S_{tmax}=80$\}$. The number at the beginning of each row indicates the number of sampling steps for the row. Each column corresponds to a single class. All images in the same column were generated starting from the same initial noise. Except for the samples generated with n_{steps}=6, there is no noticeable quality difference between the different n_{steps}.

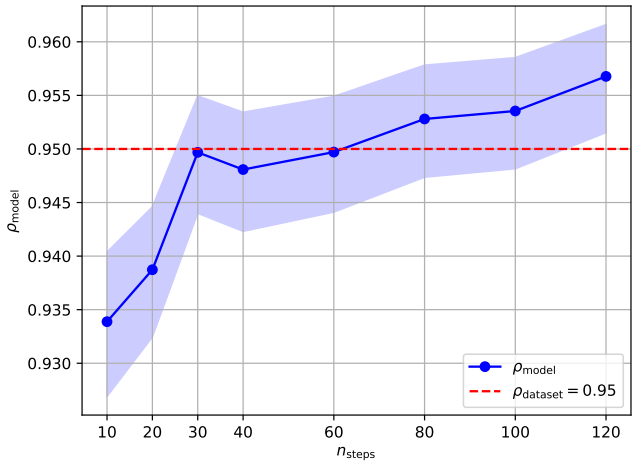


(a) ρ_{model} vs n_{steps} for a model trained on 2-classes Biased MNIST (ρ_{dataset}=0.995) using Karras stochastic sampler with hyperparameters $\{S_{\text{churn}}$=40, S_{tmin}=0.05, S_{tmax}=50$\}$. Each point is obtained by using the estimator as in Eq. (8) on 8000 generated images.

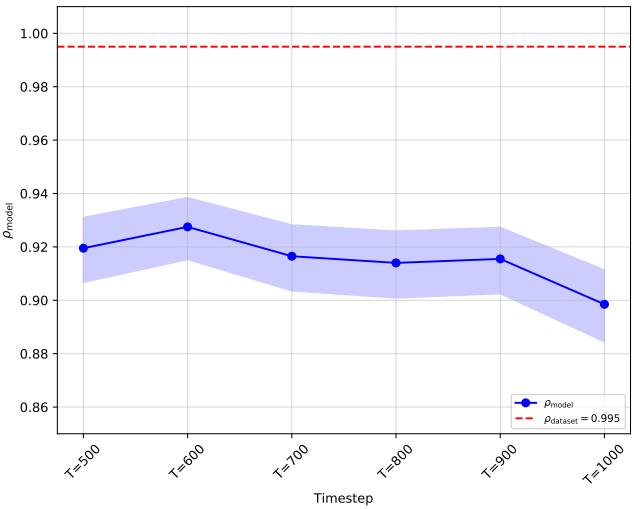


(b) ρ_{model} vs S_{tmin} for a model trained on 10-classes Biased MNIST (ρ_{dataset}=0.9) using Karras stochastic sampler with fixed hyperparameters $\{S_{\text{churn}}$=80, S_{tmax}=80$\}$ and varying n_{steps}. Same data as in Fig. 7 but presented differently.

Figure 18. Supplementary results on Biased MNIST.



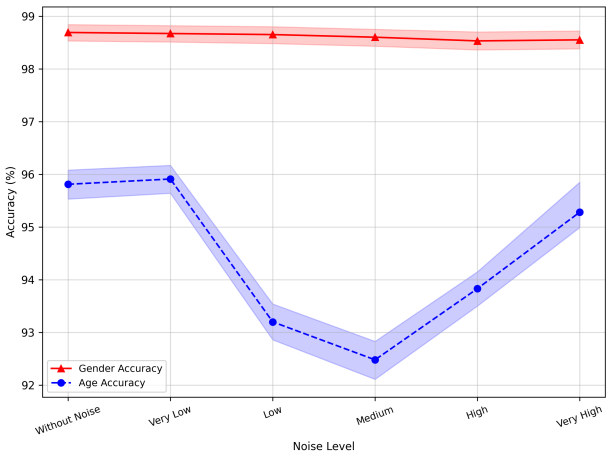
(a) ρ_{model} vs n_{steps} for a model trained on BFFHQ (ρ_{dataset}=0.95)) using Karras stochastic samplers with hyperparameters $\{S_{\text{churn}}$=80, S_{tmin}=0.01, S_{tmax}=80$\}$.



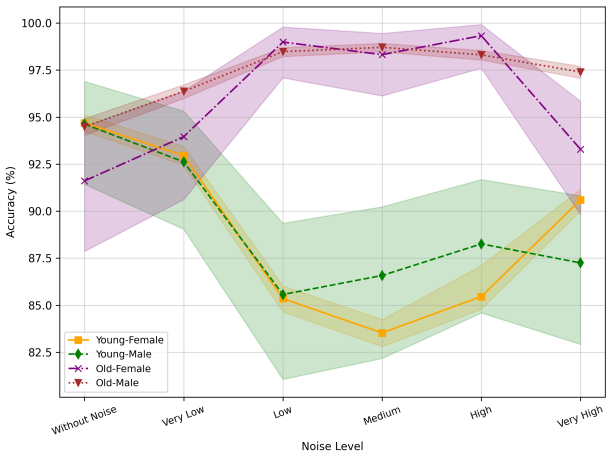
(b) ρ_{model} vs number of time steps on BFFHQ (ρ_{dataset}=0.995) using classic DDPM version. Each point is obtained by using the oracle to classify the generated images and the bias.

Figure 19. Supplementary results on BFFHQ.

Figure 20. History of the generation of 10 samples (one per row) by a model trained on 10-classes Biased MNIST (ρ_{dataset}=0.9) using Karras stochastic sampler with hyperparameters $\{n_{\text{steps}}$=60, S_{churn}=80, S_{tmin}=0.01, S_{tmax}=80$\}$. Each column corresponds to a single time step. Under the images is the RGB average of the non-white pixels (it can become negative for high noise values). Starting from $t = 15.2$, it becomes possible to guess the end color of the image for the 6 first classes by setting the highest values to 255 and the others to 0. It means that even if the images remain very noisy, the bias has already started to appear.



(a) Accuracy of the oracle in gender and age prediction for different levels of noise injected in the latent space of Stable Diffusion.



(b) Group-wise accuracy of the oracle in gender and age prediction for different levels of noise injected in the latent space of Stable Diffusion.

Figure 22. Random train samples with all considered levels of noise injected in the latent space of Stable Diffusion.

Figure 23. Examples of the images generated with CDPM. Each column corresponds to one $T \in \{500, 600, 700, 800, 900, 100\}$.

References

[1] Hyojin Bahng, Sanghyuk Chun, Sangdoo Yun, Jaegul Choo, and Seong Joon Oh. Learning de-biased representations with biased representations. In *ICML*, 2020. 1

[2] Federico Bianchi, Pratyusha Kalluri, Esin Durmus, Faisal Ladhak, Myra Cheng, Debora Nozza, Tatsunori Hashimoto, Dan Jurafsky, James Zou, and Aylin Caliskan. Easily accessible text-to-image generation amplifies demographic stereotypes at large scale. In *ACM Conference on Fairness, Accountability, and Transparency*, 2023. 1

[3] Noa Garcia, Yusuke Hirota, Yankun Wu, and Yuta Nakashima. Uncurated image-text datasets: Shedding light on demographic bias. In *CVPR*, 2023. 1

[4] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, Günter Klambauer, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a nash equilibrium. *arXiv preprint arXiv:1706.08500*, 2017. 2

[5] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020. 2

[6] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. In *NeurIPS*, 2022. 2, 3

[7] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *CVPR*, 2019. 1

[8] Zhiheng Li, Anthony Hoogs, and Chenliang Xu. Discover and mitigate unknown biases with debiasing alternate networks. In *ECCV*, 2022. 1, 3

[9] Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan LI, and Jun Zhu. Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps. In *NeurIPS*, 2022. 2, 3

[10] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. *arXiv preprint arXiv:2103.00020*, 2021. 2

[11] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev. Laion-5b: An open large-scale dataset for training next generation image-text models. In *NeurIPS*, 2022. 1

[12] Preethi Seshadri, Sameer Singh, and Yanai Elazar. The bias amplification paradox in text-to-image generation. In *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2024. 2

[13] Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020. 2, 3

[14] Xiaoshi Wu, Yiming Hao, Keqiang Sun, Yixiong Chen, Feng Zhu, Rui Zhao, and Hongsheng Li. Human preference score v2: A solid benchmark for evaluating human preferences of text-to-image synthesis. *arXiv preprint arXiv:2306.09341*, 2023. 2

[15] Yankun Wu, Yuta Nakashima, and Noa Garcia. Gender bias evaluation in text-to-image generation: A survey. *arXiv preprint arXiv:2408.11358*, 2024. 2