Some Challenges around Retraining Generative Models on their own Data

Quentin Bertrand

Équipe MALICE

Joint work with D. Ferbach, J. A. Bose, M. Jiralerspong, A. Duplessis and G. Gidel

What are Generative Models? 1/3

Data $x_1, \ldots, x_n$

Goal: new synthetic samples $\tilde{\mathbf{x}}_i$

Generative Model 101

- **Setting**: Access to samples $\underbrace{x_1, \ldots, x_n}_{\text{unlabelled data}}$

What are Generative Models? 1/3

Data $x_1, \ldots, x_n$

Goal: new synthetic samples $\tilde{\mathbf{x}}_i$

Generative Model 101

- **Setting**: Access to $\overbrace{\text{samples } x_1, \ldots, x_n}^{\text{unlabelled data}}$

  ↪ Drawn from a probability distribution p, $x_i \sim p$

What are Generative Models? 1/3

Data $x_1, \ldots, x_n$

Goal: new synthetic samples $\tilde{\mathbf{x}}_i$

Generative Model 101

- **Setting**: Access to $\overbrace{\text{samples } x_1, \ldots, x_n}^{\text{unlabelled data}}$

  $\hookrightarrow$ Drawn from a probability distribution p, $x_i \sim p$

  $\hookrightarrow$ *e.g.*, a set of images

What are Generative Models? 1/3

Data $x_1, \dots, x_n$

Goal: new synthetic samples $\tilde{\mathbf{x}}_i$

Generative Model 101

- **Setting**: Access to $\underbrace{\text{samples } x_1, \dots, x_n}^{\text{unlabelled data}}$
  - ↪ Drawn from a probability distribution p, $x_i \sim p$
  - ↪ *e.g.*, a set of images
- **Goal**: Create new samples $\tilde{\mathbf{x}}_i \sim p$

What are Generative Models? 1/3

Data $x_1, \dots, x_n$

Goal: new synthetic samples $\tilde{\mathbf{x}}_i$

Generative Model 101

- **Setting**: Access to $\overbrace{\text{samples } x_1, \dots, x_n}^{\text{unlabelled data}}$
  - $\hookrightarrow$ Drawn from a probability distribution p, $x_i \sim p$
  - $\hookrightarrow$ *e.g.*, a set of images
- **Goal**: Create new samples $\tilde{\mathbf{x}}_i \sim p$
  - $\hookrightarrow$ *e.g.*, generate images

What are Generative Models? 1/3

Data $x_1, \ldots, x_n$

Goal: new synthetic samples $\tilde{\mathbf{x}}_i$

Generative Model 101

- **Setting**: Access to $\overbrace{\text{samples } x_1, \ldots, x_n}^{\text{unlabelled data}}$
  - $\hookrightarrow$ Drawn from a probability distribution p, $x_i \sim p$
  - $\hookrightarrow$ *e.g.*, a set of images
- **Goal**: Create new samples $\tilde{\mathbf{x}}_i \sim p$
  - $\hookrightarrow$ *e.g.*, generate images

What are Generative Models? 2/3

Data $(x_1, y_1), \ldots, (x_n, y_n)$

Goal: new synthetic samples $\tilde{\mathbf{x}}_i | y_i$

Generative Model 201 (Class Conditional Generative Models)

- **Setting**: Access to samples $\overbrace{(x_1, y_1), \ldots, (x_n, y_n)}^{\text{labelled data}}$

What are Generative Models? 2/3

Data $(x_1, y_1), \dots, (x_n, y_n)$

Goal: new synthetic samples $\tilde{\mathbf{x}}_i | y_i$

Generative Model 201 (Class Conditional Generative Models)

- **Setting**: Access to $\overbrace{\text{samples } (x_1, y_1), \dots, (x_n, y_n)}^{\text{labelled data}}$

  ↪ Drawn from a probability distribution p, $x_i \sim p(\cdot | y_i)$

What are Generative Models? 2/3

Data $(x_1, y_1), \ldots, (x_n, y_n)$

Goal: new synthetic samples $\tilde{\mathbf{x}}_i | y_i$

Generative Model 201 (Class Conditional Generative Models)

- **Setting**: Access to $\overbrace{\text{samples } (x_1, y_1), \ldots, (x_n, y_n)}^{\text{labelled data}}$

  ↪ Drawn from a probability distribution p, $x_i \sim p(\cdot | y_i)$

  ↪ *e.g.*, a set of images x_i with class y_i

What are Generative Models? 2/3

Data $(x_1, y_1), \ldots, (x_n, y_n)$

Goal: new synthetic samples $\tilde{\mathbf{x}}_i | y_i$

Generative Model 201 (Class Conditional Generative Models)

- **Setting**: Access to $\overbrace{\text{samples } (x_1, y_1), \ldots, (x_n, y_n)}^{\text{labelled data}}$

  ↪ Drawn from a probability distribution p, $x_i \sim p(\cdot | y_i)$

  ↪ *e.g.*, a set of images x_i with class y_i
- **Goal**: Create new samples given a class $\tilde{\mathbf{x}}_i \sim p(\cdot | y_i)$

What are Generative Models? 2/3

Data $(x_1, y_1), \ldots, (x_n, y_n)$

Goal: new synthetic samples $\tilde{\mathbf{x}}_i | y_i$

Generative Model 201 (Class Conditional Generative Models)

- **Setting**: Access to $\overbrace{\text{samples } (x_1, y_1), \ldots, (x_n, y_n)}^{\text{labelled data}}$
  - $\hookrightarrow$ Drawn from a probability distribution p, $x_i \sim p(\cdot|y_i)$
  - $\hookrightarrow$ *e.g.*, a set of images x_i with class y_i
- **Goal**: Create new samples given a class $\tilde{\mathbf{x}}_i \sim p(\cdot|y_i)$
  - $\hookrightarrow$ *e.g.*, generate images of planes

What are Generative Models? 2/3

Data $(x_1, y_1), \ldots, (x_n, y_n)$

Goal: new synthetic samples $\tilde{\mathbf{x}}_i | y_i$

Generative Model 201 (Class Conditional Generative Models)

- **Setting**: Access to samples $\overbrace{(x_1, y_1), \ldots, (x_n, y_n)}^{\text{labelled data}}$
  - ↪ Drawn from a probability distribution p, $x_i \sim p(\cdot|y_i)$
  - ↪ *e.g.*, a set of images x_i with class y_i
- **Goal**: Create new samples given a class $\tilde{\mathbf{x}}_i \sim p(\cdot|y_i)$
  - ↪ *e.g.*, generate images of planes

What are Generative Models? 3/3

MALICE team logo

An avocado chair

A house made of sushi

Text-to-Image Models

- **Setting**: trained on samples $\overbrace{(x_1, y_1), \ldots, (x_n, y_n)}^{\text{pairs (image, caption)}}$

What are Generative Models? 3/3

MALICE team logo

An avocado chair

A house made of sushi

Text-to-Image Models

- **Setting**: trained on $\overbrace{\text{samples } (x_1, y_1), \ldots, (x_n, y_n)}^{\text{pairs (image, caption)}}$

  ↪ Drawn from a probability distribution p, $x_i \sim p(\cdot | \text{caption } y_i)$

What are Generative Models? 3/3

MALICE team logo

An avocado chair

A house made of sushi

Text-to-Image Models

- **Setting**: trained on samples $\overbrace{(x_1, y_1), \dots, (x_n, y_n)}^{\text{pairs (image, caption)}}$
  - $\hookrightarrow$ Drawn from a probability distribution p, $x_i \sim p(\cdot|\text{caption } y_i)$
- **Trick**: Create a 'meaningful' representation of the text/caption

What are Generative Models? 3/3

MALICE team logo

An avocado chair

A house made of sushi

Text-to-Image Models

- **Setting**: trained on $\overbrace{\text{samples } (x_1, y_1), \ldots, (x_n, y_n)}^{\text{pairs (image, caption)}}$

  $\hookrightarrow$ Drawn from a probability distribution p, $x_i \sim p(\cdot|\text{caption } y_i)$
- **Trick**: Create a 'meaningful' representation of the text/caption

A Few Comments

- **Text-to-Text Models**: Conditional text generation w.r.t. the previous text sequence
- **"From Noise to Structure"**

A Few Comments

- **Text-to-Text Models**: Conditional text generation w.r.t. the previous text sequence
- **"From Noise to Structure"**
- Just samples, no density

A Few Comments

- **Text-to-Text Models**: Conditional text generation w.r.t. the previous text sequence
- **"From Noise to Structure"**
- Just samples, no density

  $\hookrightarrow$ Too hard to sample from the set of natural images

A Few Comments

- **Text-to-Text Models**: Conditional text generation w.r.t. the previous text sequence
- **"From Noise to Structure"**
- Just samples, no density
  - ↪ Too hard to sample from the set of natural images
  - ↪ Easy to sample from standard Gaussian noise

A Few Comments

- **Text-to-Text Models**: Conditional text generation w.r.t. the previous text sequence
- **"From Noise to Structure"**
- Just samples, no density
  - ↪ Too hard to sample from the set of natural images
  - ↪ Easy to sample from standard Gaussian noise
  - ↪ "Rebranding of sampling"

A Few Comments

- **Text-to-Text Models**: Conditional text generation w.r.t. the previous text sequence
- **"From Noise to Structure"**
- Just samples, no density
  - ↪ Too hard to sample from the set of natural images
  - ↪ Easy to sample from standard Gaussian noise
  - ↪ "Rebranding of sampling"
- **Evaluation?** Quality of the generated images?

A Few Comments

- **Text-to-Text Models**: Conditional text generation w.r.t. the previous text sequence
- **"From Noise to Structure"**
- Just samples, no density
  - ↪ Too hard to sample from the set of natural images
  - ↪ Easy to sample from standard Gaussian noise
  - ↪ "Rebranding of sampling"
- **Evaluation?** Quality of the generated images?
  - ↪ No cross-validation

A Few Comments

- **Text-to-Text Models**: Conditional text generation w.r.t. the previous text sequence
- **"From Noise to Structure"**
- Just samples, no density
  - ↪ Too hard to sample from the set of natural images
  - ↪ Easy to sample from standard Gaussian noise
  - ↪ "Rebranding of sampling"
- **Evaluation?** Quality of the generated images?
  - ↪ No cross-validation
  - ↪ Very challenging for text-to-text models

A Few Comments

- **Text-to-Text Models**: Conditional text generation w.r.t. the previous text sequence
- **"From Noise to Structure"**
- Just samples, no density
  - ↪ Too hard to sample from the set of natural images
  - ↪ Easy to sample from standard Gaussian noise
  - ↪ "Rebranding of sampling"
- **Evaluation?** Quality of the generated images?
  - ↪ No cross-validation
  - ↪ Very challenging for text-to-text models
  - ↪ Often use other models to assess the quality *e.g.*, GPT

A Few Comments

- **Text-to-Text Models**: Conditional text generation w.r.t. the previous text sequence
- **"From Noise to Structure"**
- Just samples, no density
  - ↪ Too hard to sample from the set of natural images
  - ↪ Easy to sample from standard Gaussian noise
  - ↪ "Rebranding of sampling"
- **Evaluation?** Quality of the generated images?
  - ↪ No cross-validation
  - ↪ Very challenging for text-to-text models
  - ↪ Often use other models to assess the quality *e.g.*, GPT

Applications of Generative Models $1/2$

Until 2021, mostly Image-Based Applications, mostly GANs[a]

[a] I. Goodfellow et al. "Generative adversarial nets". In: *NeurIPS* (2014).

↪ Generate Photorealistic Images

↪ Semantic Segmentation[a]

↪ Image-to-Image (Inpainting, Denoising, Style Transfer)

↪ Text-to-Image[b]

[a] P. Luc et al. "Semantic segmentation using adversarial networks". In: *arXiv preprint arXiv:1611.08408* (2016).

[b] H. Zhang et al. "StackGAN: Text to photo-realistic image synthesis with stacked generative adversarial networks". In: *ICCV*. 2017.

Applications of Generative Models $2/2$

More Recent Applications

- Large Langage Models[a] (Chat GPT)
- Text-to-Image[b] (Stable Diffusion)
- Protein Generation[c][d] (Graphs)
- Data augmentation[e]

[a] J. Achiam et al. "Gpt-4 technical report". In: *arXiv preprint arXiv:2303.08774* (2023).

[b] Stability AI. https://stability.ai/stablediffusion. Version Stable Diffusion XL. Accessed: 2023-09-09. 2023.

[c] J. L. Watson et al. "De novo design of protein structure and function with RFdiffusion". In: *Nature* 620 (2023).

[d] A. J. Bose et al. "SE(3)-Stochastic Flow Matching for Protein Backbone Generation". In: *ICLR* (2023).

[e] Z. Wang et al. "Better diffusion models further improve adversarial training". In: *ICML*. 2023.

Reasons of the Success of Generative Models

$$\text{Deep generative models} = \underbrace{\text{Compute}}_{\text{GPU}} + \underbrace{\text{Algorithms}}_{e.g.,\text{Diffusion}} + \underbrace{\text{Data}}_{\text{Web Scrapping}}$$

Generative Models Everywhere

- Powerful generative models (Diffusion, Flow Matching)
- Easy access (Midjourney, Stablediffusion, DALL·E)
- Populates the WEB with **synthetically generated images**

Inevitably Train on Synthetic Data

The Lion[1] dataset already contains synthetically generated images[2]

Hi-fidelity synthetic imagery

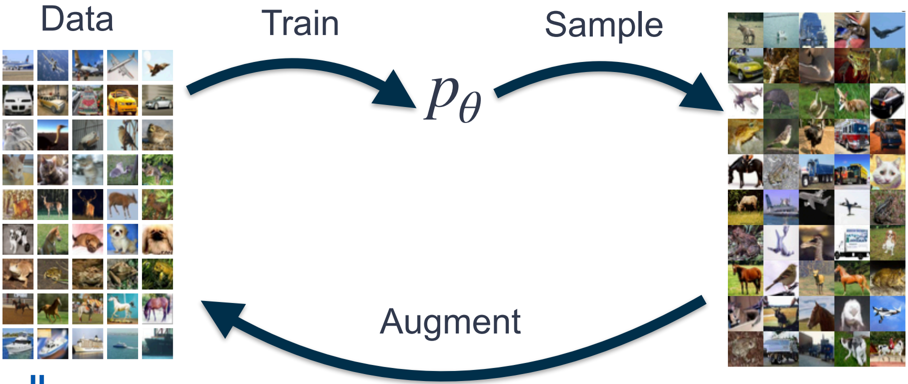
[1] C. Schuhmann et al. "Laion-5b: An open large-scale dataset for training next generation image-text models". In: *NeurIPS* (2022).

[2] S. Alemohammad et al. "Self-Consuming Generative Models Go MAD". In: (2023). arXiv: 2307.01850 [cs.LG].

What about training Generative models on their own data?

Mila

CIFAR



Data

Train

p_θ

Sample

Augment

Université de Montréal

Training on Synthetic Data, Good or Bad?

Iterative Retraining is **Bad**

- The **curse of recursion**: Training on generated data makes models forget[a]
- Self-Consuming Generative Models **MAD**[b]

[a] I. Shumailov et al. "The Curse of Recursion: Training on Generated Data Makes Models Forget". In: (2023). arXiv: 2305.17493 [cs.LG].

[b] S. Alemohammad et al. "Self-Consuming Generative Models Go MAD". In: (2023). arXiv: 2307.01850 [cs.LG].

Training on Synthetic Data, Good or Bad?

Iterative Retraining is **Bad**

- The **curse of recursion**: Training on generated data makes models forget[a]
- Self-Consuming Generative Models **MAD**[b]

[a] I. Shumailov et al. "The Curse of Recursion: Training on Generated Data Makes Models Forget". In: (2023). arXiv: 2305.17493 [cs.LG].

[b] S. Alemohammad et al. "Self-Consuming Generative Models Go MAD". In: (2023). arXiv: 2307.01850 [cs.LG].

Will generative models collapse?!

Training on Synthetic Data, Good or Bad?

Iterative Retraining is **Bad**

- The **curse of recursion**: Training on generated data makes models forget[a]
- Self-Consuming Generative Models **MAD**[b]

[a] I. Shumailov et al. "The Curse of Recursion: Training on Generated Data Makes Models Forget". In: (2023). arXiv: 2305.17493 [cs.LG].

[b] S. Alemohammad et al. "Self-Consuming Generative Models Go MAD". In: (2023). arXiv: 2307.01850 [cs.LG].

Will generative models collapse?!

Training on Synthetic Data is **Good**

- Data augmentation for downstream tasks
  - ↪ Adversarial training[a]
  - ↪ Classification with imbalanced datasets[b]
  - ↪ Generative modelling: improves performances for LLMs[c]

[a] Z. Wang et al. "Better diffusion models further improve adversarial training". In: *ICML*. 2023.

[b] R. A. Hemmat et al. "Feedback-guided Data Synthesis for Imbalanced Classification". In: *arXiv preprint arXiv:2310.00158* (2023).

Training on Synthetic Data, Good or Bad?

Iterative Retraining is **Bad**

- The **curse of recursion**: Training on generated data makes models forget[a]
- Self-Consuming Generative Models **MAD**[b]

[a] I. Shumailov et al. “The Curse of Recursion: Training on Generated Data Makes Models Forget”. In: (2023). arXiv: 2305.17493 [cs.LG].

[b] S. Alemohammad et al. “Self-Consuming Generative Models Go MAD”. In: (2023). arXiv: 2307.01850 [cs.LG].

Will generative models collapse?!

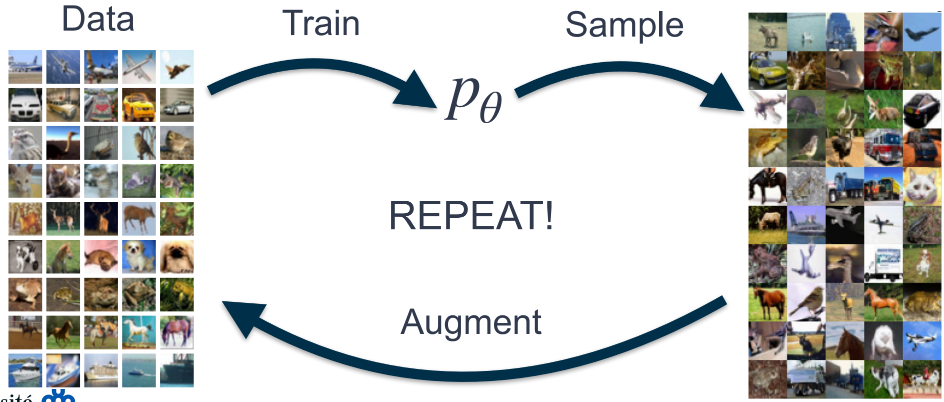
Training on Synthetic Data is **Good**

- Data augmentation for downstream tasks
  - ↪ Adversarial training[a]
  - ↪ Classification with imbalanced datasets[b]
  - ↪ Generative modelling: improves performances for LLMs[c]

[a] Z. Wang et al. “Better diffusion models further improve adversarial training”. In: *ICML*. 2023.

[b] R. A. Hemmat et al. “Feedback-guided Data Synthesis for Imbalanced Classification”. In: *arXiv preprint arXiv:2310.00158* (2023).

Iterative retraining



Data

Train

p_θ

Sample

REPEAT!

Augment

Setting

Notation

- $\hat{p}_{\text{data}}$ Empirical data distribution
- n Data points
- θ^n Parameters of the model
- p_θ Likelihood of the model

Iterative Retraining

$$\theta_0^n \in \underset{\theta' \in \Theta}{\arg\max}\, \mathbb{E}_{x \sim \hat{p}_{\text{data}}}[\log p_{\theta'}(x)]$$

Setting

Notation

- $\hat{p}_{\text{data}}$ Empirical data distribution
- n Data points
- θ^n Parameters of the model
- p_θ Likelihood of the model

Iterative Retraining

$$\theta_0^n \in \underset{\theta' \in \Theta}{\arg\max}\, \mathbb{E}_{x \sim \hat{p}_{\text{data}}} [\log p_{\theta'}(x)]$$

$$\theta_{t+1}^n \in \underbrace{\mathbb{E}_{x \sim \hat{p}_{\text{data}}} \log p_{\theta'}(x)}_{\text{Real data}} + \lambda \underbrace{\mathbb{E}_{\tilde{\mathbf{x}} \sim p_{\theta_t^n}} \log p_{\theta'}(\tilde{\mathbf{x}})}_{\text{Synthetic data}}$$

Setting

Notation

- $\hat{p}_{\text{data}}$ Empirical data distribution
- n Data points
- θ^n Parameters of the model
- p_θ Likelihood of the model

Iterative Retraining

$$\theta_0^n \in \arg\max_{\theta' \in \Theta} \mathbb{E}_{x \sim \hat{p}_{\text{data}}} [\log p_{\theta'}(x)]$$

$$\theta_{t+1}^n \in \arg\max_{\theta' \in \Theta} \underbrace{\mathbb{E}_{x \sim \hat{p}_{\text{data}}} \log p_{\theta'}(x)}_{\text{Real data}} + \lambda \underbrace{\mathbb{E}_{\tilde{\mathbf{x}} \sim p_{\theta_t^n}} \log p_{\theta'}(\tilde{\mathbf{x}})}_{\text{Synthetic data}}$$

Practical Algorithm

Algorithm: Iterative Retraining of Generative Models

```
input :  $\mathcal{D}_{\text{real}} := \{x_i\}_{i=1}^{n}$, $\mathcal{A}$  // True data, learning procedure
param:  $n_{\text{retrain.}}$, $\lambda$  // Number of retraining, proportion of gen.  data
$p_{\theta_0} = \mathcal{A}(\mathcal{D}_{\text{real}})$  // Learn generative model on true data
for $t$ in $1, \ldots, n_{\text{retrain.}}$ do
    $\mathcal{D}_{\text{synth}} = \{\tilde{\mathbf{x}}_i\}_{i=1}^{\lfloor \lambda \cdot n \rfloor}$, with $\tilde{\mathbf{x}}_i \sim p_{\theta_{t-1}}$  // Sample $\lfloor \lambda \cdot n \rfloor$ synth.  data points
    $p_{\theta_t} = \mathcal{A}(\mathcal{D}_{\text{real}} \cup \mathcal{D}_{\text{synth}})$  // Learn gen.  model on synth.  and true data
return $p_{\theta_{n_{\text{retrain.}}}}$
```

Warm Up: Only Retrain on your Own Data $1/3$

Iterative Retraining

$$\theta_0^{n} \in \underset{\theta' \in \Theta}{\arg\max}\, \mathbb{E}_{x\sim\hat{p}_{\text{data}}} [\log p_{\theta'}(x)]$$

$$\theta_{t+1}^{n} \in \underset{\theta' \in \Theta}{\arg\max}\, \underbrace{\cancel{\mathbb{E}_{x\sim\hat{p}_{\text{data}}} \log p_{\theta'}(x)}}_{\text{Real data}} + \text{✗}\, \underbrace{\mathbb{E}_{\tilde{\mathbf{x}}\sim p_{\theta_t^n}} \log p_{\theta'}(\tilde{\mathbf{x}})}_{\text{Synthetic data}}$$

Q: What will happen?

Warm Up: Only Retrain on your Own Data $1/3$

Iterative Retraining

$$\theta_0^n \in \arg\max_{\theta' \in \Theta} \mathbb{E}_{x\sim\hat{p}_{\mathrm{data}}}[\log p_{\theta'}(x)]$$

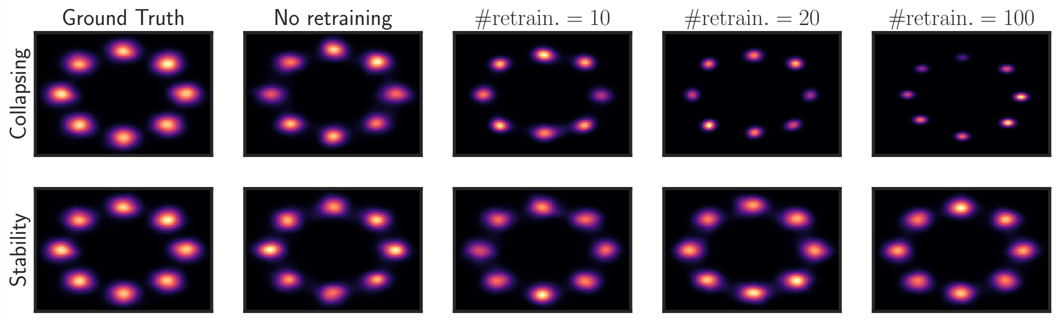
$$\theta_{t+1}^n \in \arg\max_{\theta'\in\Theta} \underbrace{\cancel{\mathbb{E}_{x\sim\hat{p}_{\mathrm{data}}} \log p_{\theta'}(x)}}_{\text{Real data}} + \text{✗}\,\underbrace{\mathbb{E}_{\tilde{\mathbf{x}}\sim p_{\theta_t^n}} \log p_{\theta'}(\tilde{\mathbf{x}})}_{\text{Synthetic data}}$$

Q: What will happen?

Warm Up: Only Retrain on your Own Data $2/3$

Q: What will happen?

A: Mode Collapse



Warm Up: Only Retrain on your Own Data $3/3$

Single unidimensional Gaussian, unbiaissed estimator

Initialization: $\mu_0,\ \sigma_0$

Data: $X_j^0 = \mu_0 + \sigma_0 Z_j$, with $Z_j \overset{\text{i.i.d.}}{\sim} \mathcal{N}_{0,1},\ 1 \leq j \leq n$

Learning step: $\begin{cases} \mu_{t+1} & = \frac{1}{n} \sum\limits_j X_j^t \\ \sigma_{t+1}^2 & = \frac{1}{n-1} \sum\limits_j \left(X_j^t - \mu_{t+1} \right)^2 \end{cases}$

Sampling step: $\left\{ X_j^{t+1} = \mu_{t+1} + \sigma_{t+1} Z_j^{t+1} \right.$, with $Z_j^{t+1} \overset{\text{i.i.d.}}{\sim} \mathcal{N}_{0,1},\ 1 \leq j \leq n$

Result

$$\mathbb{E}(\sigma_t) \leq \alpha^t \sigma_0 \underset{t \to +\infty}{\longrightarrow} 0,\ 0 \leq \alpha < 1$$

Warm Up: Only Retrain on your Own Data $3/3$

Single unidimensional Gaussian, unbiaissed estimator

Initialization: $\mu_0,\ \sigma_0$

Data: $X_j^0 = \mu_0 + \sigma_0 Z_j,$ with $Z_j \overset{\text{i.i.d.}}{\sim} \mathcal{N}_{0,1},\ 1 \le j \le n$

Learning step: $$\begin{cases} \mu_{t+1} &= \frac{1}{n} \sum_j X_j^t \\ \sigma_{t+1}^2 &= \frac{1}{n-1} \sum_j \left(X_j^t - \mu_{t+1} \right)^2 \end{cases}$$

Sampling step: $\left\{ X_j^{t+1} = \mu_{t+1} + \sigma_{t+1} Z_j^{t+1},\right.$ with $Z_j^{t+1} \overset{\text{i.i.d.}}{\sim} \mathcal{N}_{0,1},\ 1 \le j \le n$

Result

$$\mathbb{E}(\sigma_t) \le \alpha^t \sigma_0 \underset{t \to +\infty}{\longrightarrow} 0,\ 0 \le \alpha < 1$$

Same type of results holds for a single multidimensional Gaussian

Warm Up: Only Retrain on your Own Data $3/3$

Single unidimensional Gaussian, unbiaissed estimator

Initialization: $\mu_0,\ \sigma_0$

Data: $X_j^0 = \mu_0 + \sigma_0 Z_j$, with $Z_j \overset{\text{i.i.d.}}{\sim} \mathcal{N}_{0,1},\ 1 \leq j \leq n$

Learning step: $$\begin{cases} \mu_{t+1} &= \frac{1}{n}\sum_j X_j^t \\ \sigma_{t+1}^2 &= \frac{1}{n-1}\sum_j \left(X_j^t - \mu_{t+1}\right)^2 \end{cases}$$

Sampling step: $\left\{ X_j^{t+1} = \mu_{t+1} + \sigma_{t+1} Z_j^{t+1} \right.$, with $Z_j^{t+1} \overset{\text{i.i.d.}}{\sim} \mathcal{N}_{0,1},\ 1 \leq j \leq n$

Result

$$\mathbb{E}(\sigma_t) \leq \alpha^t \sigma_0 \underset{t \to +\infty}{\longrightarrow} 0,\ 0 \leq \alpha < 1$$

Same type of results holds for a single multidimensional Gaussian

General Case

Iterative Retraining

$$\theta_0^n \in \underset{\theta' \in \Theta}{\arg\max}\, \mathbb{E}_{x \sim \hat{p}_{\mathrm{data}}} [\log p_{\theta'}(x)]$$

$$\theta_{t+1}^n \in \underset{\theta' \in \Theta}{\arg\max}\, \underbrace{\mathbb{E}_{x \sim \hat{p}_{\mathrm{data}}} \log p_{\theta'}(x)}_{\text{Real data}} + \lambda \underbrace{\mathbb{E}_{\tilde{\mathbf{x}} \sim p_{\theta_t^n}} \log p_{\theta'}(\tilde{\mathbf{x}})}_{\text{Synthetic data}} := \mathcal{G}(\theta_t^n)$$

Idea

- Fixed-point iteration $\theta_{t+1}^n = \mathcal{G}(\theta_t^n)$
- Study the stability of the fixed-point iteration
- Link with performative prediction!

General Case

Iterative Retraining

$$\theta_0^n \in \underset{\theta' \in \Theta}{\arg\max}\, \mathbb{E}_{x \sim \hat{p}_{\mathrm{data}}} [\log p_{\theta'}(x)]$$

$$\theta_{t+1}^n \in \underset{\theta' \in \Theta}{\arg\max}\, \underbrace{\mathbb{E}_{x \sim \hat{p}_{\mathrm{data}}} \log p_{\theta'}(x)}_{\text{Real data}} + \lambda \underbrace{\mathbb{E}_{\tilde{\mathbf{x}} \sim p_{\theta_t^n}} \log p_{\theta'}(\tilde{\mathbf{x}})}_{\text{Synthetic data}} := \mathcal{G}(\theta_t^n)$$

Idea

- Fixed-point iteration $\theta_{t+1}^n = \mathcal{G}(\theta_t^n)$
- Study the stability of the fixed-point iteration
- Link with performative prediction!

Retrain of Generative Models: Informal

Assumptions

- Regularity of the log-likelihood
  - $\hookrightarrow$ Local Lipschitzness and strong convexity
- The first generative model is "good enough"
  - $\hookrightarrow$ $\mathcal{W}(p_{\text{data}}, p_{\theta_0}) < \epsilon$
- Infinite Data

Result

- Regularity + good enough model + infinite data
- $\implies$ stability of the fixed-point $\mathcal{G}(\theta)$

Retrain of Generative Models: Informal

Assumptions

- Regularity of the log-likelihood

  $\hookrightarrow$ Local Lipschitzness and strong convexity
- The first generative model is "good enough"

  $\hookrightarrow$ $\mathcal{W}(p_{\text{data}}, p_{\theta_0}) < \epsilon$
- Infinite Data

Result

- Regularity + good enough model + infinite data
- $\implies$ stability of the fixed-point $\mathcal{G}(\theta)$

- Can be extended with finite sample
- Requires extra sample complexity assumption

Retrain of Generative Models: Informal

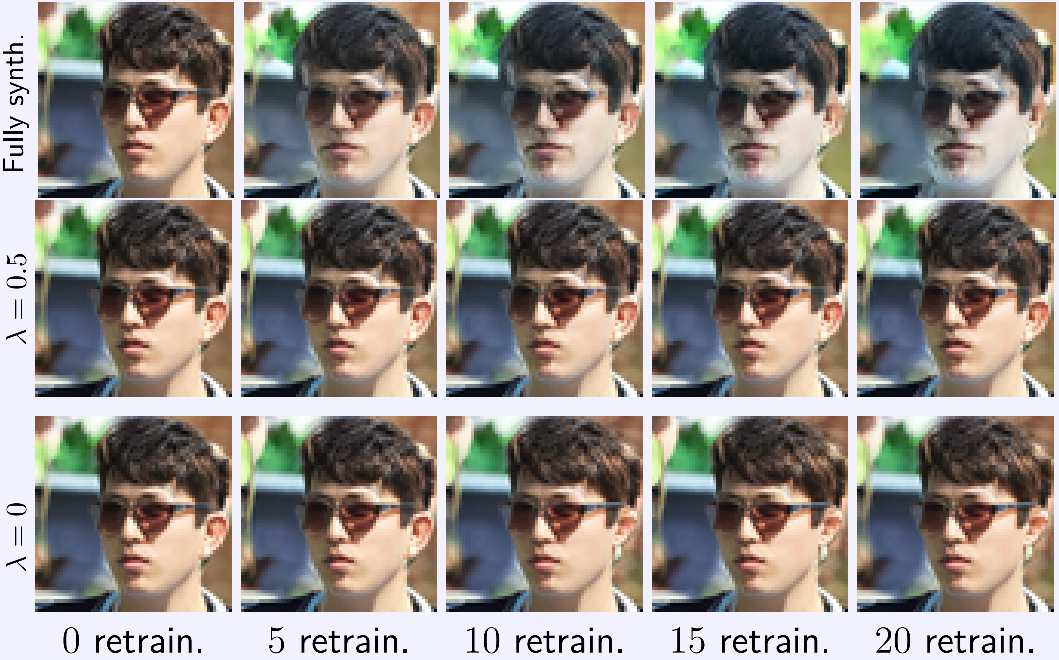
Assumptions

- Regularity of the log-likelihood
  - $\hookrightarrow$ Local Lipschitzness and strong convexity
- The first generative model is "good enough"
  - $\hookrightarrow$ $\mathcal{W}(p_{\text{data}}, p_{\theta_0}) < \epsilon$
- Infinite Data

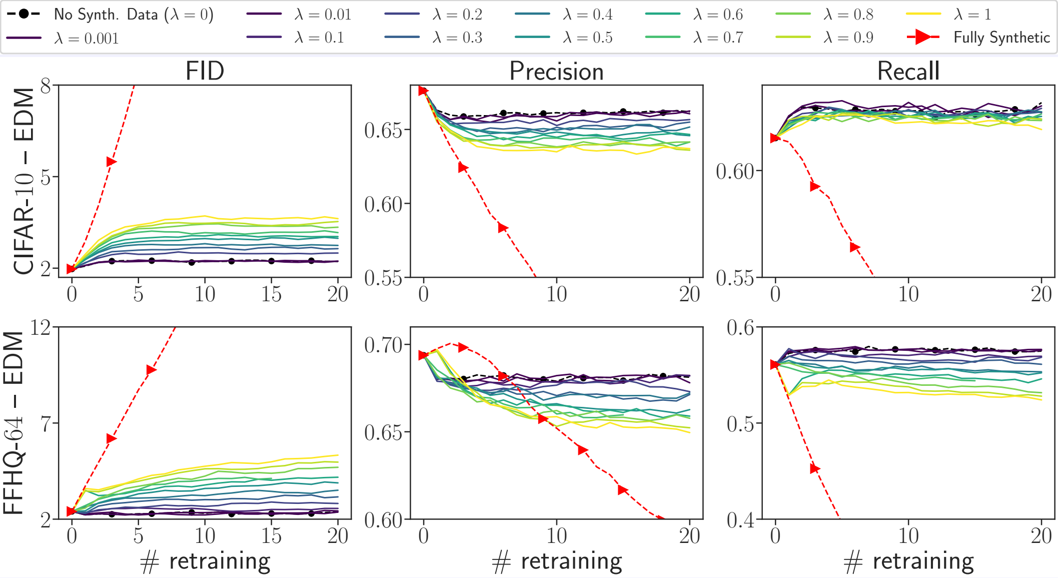
Result

- Regularity + good enough model + infinite data
- $\implies$ stability of the fixed-point $\mathcal{G}(\theta)$

- Can be extended with finite sample
- Requires extra sample complexity assumption


Fully synth.
λ = 0.5
λ = 0
0 retrain.
5 retrain.
10 retrain.
15 retrain.
20 retrain.

Experiments



Conclusion and Future Work

Future Work

- Data augmentation? → Filtering Procedure
  - ↪ Score for each samples? Downstream-task specific?
    - ↪ Feature Likelihood Score (FLS)[a]
    - ↪ Classifier to score the samples[b]
      - ↪ Correlation between accuracy and sample quality?
  - ↪ Theory?
- Links with reinforcement learning / semi-supervised learning[c]

[a] M. Jiralerspong et al. "Feature Likelihood Score: Evaluating Generalization of Generative Models Using Samples". In: *NeurIPS* (2023).

[b] R. A. Hemmat et al. "Feedback-guided Data Synthesis for Imbalanced Classification". In: *arXiv preprint arXiv:2310.00158* (2023).

[c] D. Ferbach et al. "Self-Consuming Generative Models with Curated Data Provably Optimize Human Preferences". In: *arXiv preprint arXiv:2407.09499* (2024).

Thank You!

Conclusion and Future Work

Future Work

- Data augmentation? → Filtering Procedure
  - ↪ Score for each samples? Downstream-task specific?
    - ↪ Feature Likelihood Score (FLS)[a]
    - ↪ Classifier to score the samples[b]
      - ↪ Correlation between accuracy and sample quality?
  - ↪ Theory?
- Links with reinforcement learning / semi-supervised learning[c]

[a] M. Jiralerspong et al. “Feature Likelihood Score: Evaluating Generalization of Generative Models Using Samples”. In: *NeurIPS* (2023).

[b] R. A. Hemmat et al. “Feedback-guided Data Synthesis for Imbalanced Classification”. In: *arXiv preprint arXiv:2310.00158* (2023).

[c] D. Ferbach et al. “Self-Consuming Generative Models with Curated Data Provably Optimize Human Preferences”. In: *arXiv preprint arXiv:2407.09499* (2024).

Thank You!

- Achiam, J. et al. “Gpt-4 technical report”. In: *arXiv preprint arXiv:2303.08774* (2023).
- Alemohammad, S. et al. “Self-Consuming Generative Models Go MAD”. In: (2023). arXiv: `2307.01850 [cs.LG]`.
- Bose, A. J. et al. “SE(3)-Stochastic Flow Matching for Protein Backbone Generation”. In: *ICLR* (2023).
- Ferbach, D. et al. “Self-Consuming Generative Models with Curated Data Provably Optimize Human Preferences”. In: *arXiv preprint arXiv:2407.09499* (2024).
- Goodfellow, I. et al. “Generative adversarial nets”. In: *NeurIPS* (2014).
- Gulcehre, C. et al. “Reinforced self-training (REST) for language modeling”. In: (2023). arXiv: `2308.08998 [cs.CL]`.
- Hemmat, R. A. et al. “Feedback-guided Data Synthesis for Imbalanced Classification”. In: *arXiv preprint arXiv:2310.00158* (2023).
- Jiralerspong, M. et al. “Feature Likelihood Score: Evaluating Generalization of Generative Models Using Samples”. In: *NeurIPS* (2023).
- Luc, P. et al. “Semantic segmentation using adversarial networks”. In: *arXiv preprint arXiv:1611.08408* (2016).

- Schuhmann, C. et al. "Laion-5b: An open large-scale dataset for training next generation image-text models". In: *NeurIPS* (2022).
- Shumailov, I. et al. "The Curse of Recursion: Training on Generated Data Makes Models Forget". In: (2023). arXiv: 2305.17493 [cs.LG].
- Stability AI. https://stability.ai/stablediffusion. Version Stable Diffusion XL. Accessed: 2023-09-09. 2023.
- Wang, Z. et al. "Better diffusion models further improve adversarial training". In: *ICML*. 2023.
- Watson, J. L. et al. "De novo design of protein structure and function with RFdiffusion". In: *Nature* 620 (2023).
- Zhang, H. et al. "StackGAN: Text to photo-realistic image synthesis with stacked generative adversarial networks". In: *ICCV*. 2017.