SBM 1: Score-based Models

CS 598: Deep Generative and Dynamical Models

Instructor: Arindam Banerjee

November 11, 2021

Motivation

- Classical approaches based on normalized models

$$p(\xi, \theta) = \frac{1}{Z(\theta)} q(\xi; \theta)$$

- The normalization involves high-d integration

$$Z(\theta) = \int_{\xi} q(\xi; \theta) d\xi$$

- Classical approaches based on MCMC or variational inference
- Difficult to scale up to high-dimensions
- Can we build valid models without using $Z(\theta)$

Estimation by Score Matching

- Consider parametric model family $\psi(\xi;\theta)$
- Score function based on gradient of log-likelihood w.r.t. location ξ, rather than model parameter θ:

$$\psi(\xi;\theta) = \nabla_\xi \log p_\theta(\xi) = \begin{bmatrix} \frac{\partial \log p(\xi;\theta)}{\partial \xi_1} \\ \vdots \\ \frac{\partial \log p(\xi;\theta)}{\partial \xi_p} \end{bmatrix} = \begin{bmatrix} \psi_1(\xi;\theta) \\ \vdots \\ \psi_p(\xi;\theta) \end{bmatrix}$$

- Can be computed for an assumed form of $\psi(\xi;\theta)$
- Does not depend on $Z(\theta)$ since $\nabla_\xi Z(\theta) = 0$
- For data distribution P_x, score

$$\psi_x(\xi) = \nabla_\xi \log p_x(\xi)$$

- Challenge: Need to know $p_x(\xi)$ over all ξ

Main Result: Tractable Score Matching

- Goal: minimize the following objective function

$$J(\theta) = \frac{1}{2}\int_{\xi} p_{\text{x}}(\xi)\|\psi(\xi;\theta) - \psi_x(\xi)\|^2 d\xi$$

- Assume $\psi(\xi;\theta)$ is differentiable and satisfies some regularity conditions
- **Theorem:** Objective function can be expressed as

$$J(\theta) = \mathbb{E}_{P_{\text{x}}}\left[\sum_{i=1}^{p}\left\{\frac{\partial^2 \log p(\xi;\theta)}{\partial \xi_i^2} + \frac{1}{2}\left(\frac{\partial \log p(\xi;\theta)}{\partial \xi_i}\right)^2\right\}\right] + c$$

$$= \int_{\xi}\left(\text{Tr}(\nabla\psi(\xi;\theta)) + \frac{1}{2}\|\psi(\xi;\theta)\|^2\right) p_x(\xi) d\xi + c$$

- Only need to compute $\mathbb{E}_{P_{\text{x}}}$, not gradients $\nabla_x \log p_x(\xi)$

Finite Sample Objective Function

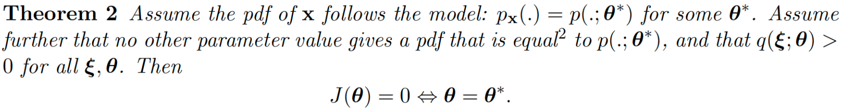
- Replace expectation with average over finite samples $\{x_i\}_{i=1}^n$

$$\bar{J}(\theta) = \frac{1}{n}\sum_{i=1}^{n}\left(\text{Tr}(\nabla\psi(x_i;\theta)) + \frac{1}{2}\|\psi(x_i;\theta)\|^2\right)$$

- Asymptotically equivalent to J
- In practice, (nonconvex) optimization problem in θ

Consistency

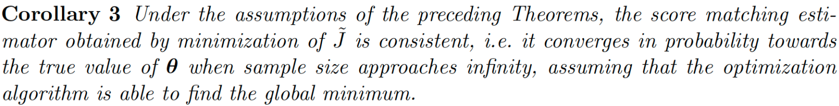
- If the model is non-degenerate, we have local consistency



Theorem 2 *Assume the pdf of* $\mathbf{x}$ *follows the model:* $p_{\mathbf{x}}(.) = p(.;\boldsymbol{\theta}^*)$ *for some* $\boldsymbol{\theta}^*$. *Assume further that no other parameter value gives a pdf that is equal*[2] *to* $p(.;\boldsymbol{\theta}^*)$, *and that* $q(\boldsymbol{\xi};\boldsymbol{\theta}) > 0$ *for all* $\boldsymbol{\xi},\boldsymbol{\theta}$. *Then*

$$J(\boldsymbol{\theta}) = 0 \Leftrightarrow \boldsymbol{\theta} = \boldsymbol{\theta}^*.$$

- Estimation based on $\bar{J}$ converges in probability



Corollary 3 *Under the assumptions of the preceding Theorems, the score matching estimator obtained by minimization of* $\tilde{J}$ *is consistent, i.e. it converges in probability towards the true value of* $\boldsymbol{\theta}$ *when sample size approaches infinity, assuming that the optimization algorithm is able to find the global minimum.*

Example: Multivariate Gaussian

- Parameterized by mean μ and precision $M = \Sigma^{-1}$
$$p(\mathrm{x}; \mu, M) = \frac{1}{Z(M)} \exp\left(-\frac{1}{2}(\mathrm{x} - \mu)^T M(x - \mu)\right)$$

- The finite sample objective
$$\bar{J}(\mu, M) = \frac{1}{n}\sum_{i=1}^{n}\left(-\operatorname{Tr}(M) + \frac{1}{2}(\mathrm{x}_i - \mu)^T MM(x_i - \mu)\right)$$

Example: Multivariate Gaussian (Contd.)

- Gradient w.r.t. μ

$$\nabla_\mu \bar{J} = MM\mu - MM\frac{1}{n}\sum_{i=1}^{n} \mathrm{x}_i$$

- Gradient w.r.t. M

$$\nabla_M \bar{J} = -I + M\frac{1}{n}\sum_{i=1}^{n}(\mathrm{x}_i - \mu)(\mathrm{x}_i - \mu)^T + \left(\frac{1}{n}\sum_{i=1}^{n}(\mathrm{x}_i - \mu)(\mathrm{x}_i - \mu)^T\right) M$$

- Solution is exactly the same as maximum likelihood estimation

Example: Independent Component Analysis (ICA)

- Basic form of the ICA model

$$\log p(\mathrm{x}; W) = \sum_{k=1}^{p} G(\mathrm{w}_k^T \mathrm{x}) + Z(w_1, \dots, w_p)$$

    - Normalization constant $-\log|\det W|$, W has rows w_k
- Generative model: $s_k, k = 1, \dots, p$ are i.i.d. distributed as $\exp(G(s_k))$

$$\mathrm{x} = As \ , \qquad A = W^{-1}$$

    - $p(\mathrm{x})$ is the distribution of x
- Components s_k follow the logistic distribution $\exp(G(s_k))$ with

$$G(s) = -2 \log cosh\left(\frac{\pi}{2\sqrt{3}} s\right) - \log 4$$

- Score-based model based on $\nabla_W \log p(\mathrm{x}; W)$, with $g(s) = G'(s)$

$$\psi(\mathrm{x}; W) = \sum_{k=1}^{p} \mathrm{w}_k g(\mathrm{w}_k^T \mathrm{x})$$

Results: ICA

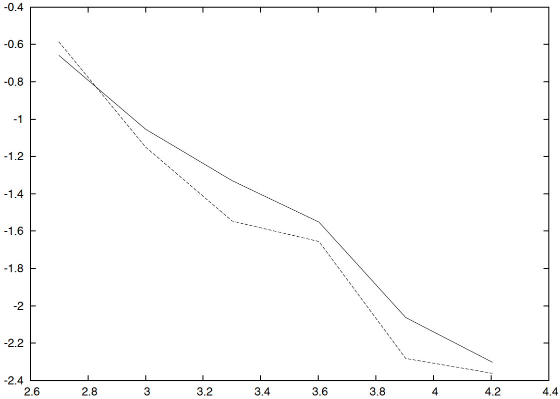


Figure 1: The estimation errors of score matching (solid line) compared with errors of maximum likelihood estimation (dashed line) for the basic ICA model. Horizontal axis: $\log_{10}$ of sample size. Vertical axis: $\log_{10}$ of estimation error.

Results: ICA with Misspecification

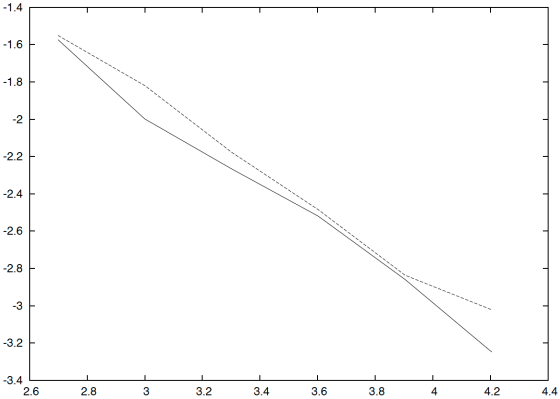


Figure 2: The estimation errors of score matching compared with errors of maximum likelihood estimation for the basic ICA model. This time, the pdf of the independent components was slightly misspecified. Legend as in Fig. 1.

Example: Overcomplete ICA

- Number of components m is larger than the data dimensionality p
- Log-likelihood is given by
$$\log p(\mathrm{x}) = \sum_{k=1}^{m} \alpha_k G(\mathrm{w}_k^T \mathrm{x}) + Z(\mathrm{w}_{1:m}, \alpha_{1:m})$$
- The vectors $\mathrm{w}_k = (w_{k1}, \ldots, w_{kp})$ satisfy $\|\mathrm{w}_k\|_2 = 1$
- α_k handles different distributions for different projections $\mathrm{w}_k^T \mathrm{x}$
- Score function
$$\psi(\mathrm{x}; W, \alpha_{1:m}) = \sum_{k=1}^{m} \alpha_k w_k g(\mathrm{w}_k^T \mathrm{x})$$

Results: Overcomplete ICA, Estimated Filters

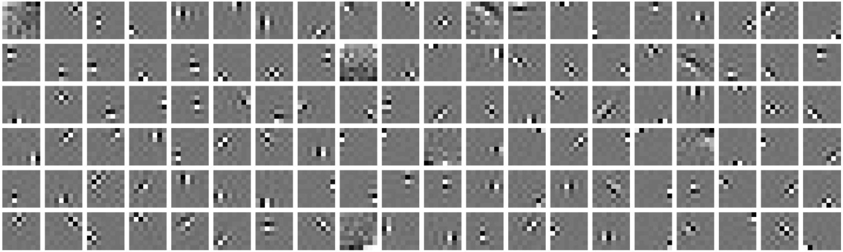


Figure 3: The overcomplete set of filters $\mathbf{w}_i$ estimated from natural image data. Note that no dimension reduction was performed, and we show filters instead of basis vectors, which is why the results are much less smooth and "beautiful" than some published ICA results (Hyvärinen et al., 2001).

Results: Inner Product Distribution

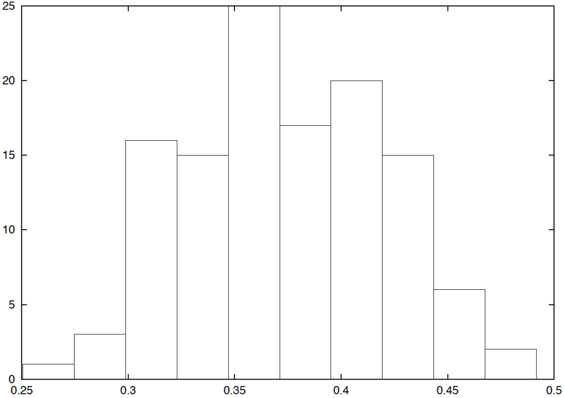


Figure 4: The distribution of maximal dot-products of a filter $\mathbf{w}_i$ with all other filters, computed in the whitened space.

Score-based Model

- Score $\nabla_x \log p(x)$, samples $\{\mathrm{x}_i\}_{i=1}^n, \mathrm{x}_i \in \mathbb{R}^D$
- Score network $s_\theta : \mathbb{R}^D \mapsto \mathbb{R}^D$ to approximate score pf $p(\mathrm{x})$
- Recall that

$$\mathbb{E}_{p(\mathrm{x})}\left[\mathrm{Tr}(\nabla_x s_\theta(\mathrm{x})) + \frac{1}{2}\|s_\theta(\mathrm{x})\|_2^2\right]$$

- Arguably not scalable for high-d due to the $\mathrm{Tr}(\nabla_x s_\theta(\mathrm{x}))$ term
- Sliced score matching by efficiently estimating trace

$$\mathbb{E}_{\mathrm{v}\sim\mathcal{N}(0,\mathbb{I})}\mathbb{E}_{p(\mathrm{x})}\left[\mathrm{v}^T\nabla_x s_\theta(\mathrm{x})\mathrm{v} + \frac{1}{2}\|s_\theta(\mathrm{x})\|_2^2\right]$$

Score-Matching with Noise

- Completely circumvent the $\mathrm{Tr}(\nabla_x s_\theta(\mathrm{x}))$ term
- Perturb data x with a pre-specified distribution $q_\sigma(\tilde{x}|x)$
- Objective shown to be

$$\frac{1}{2}\mathbb{E}_{q_\sigma(\tilde{\mathrm{x}}|\mathrm{x})p(\mathrm{x})}\left[\left\|s_\theta(\mathrm{x}) - \nabla_{\tilde{x}} \log q_\sigma(\tilde{x}|\mathrm{x})\right\|^2\right]$$

- The optimal network $s_{\theta^*}(\mathrm{x}) = \nabla_{\mathrm{x}} \log q_\sigma(\mathrm{x})$
- Close to the true score $\nabla \log p(\mathrm{x})$ when $q_\sigma(x) \approx p(\mathrm{x})$

Sampling with Langevin Dynamics

- Sampling from $p(\mathrm{x})$ using score function $\nabla_{\mathrm{x}} \log p(\mathrm{x})$
- Sample $\tilde{x}_0 \sim \pi$ (prior), for ‘small’ step-size ϵ

$$\tilde{\mathrm{x}}_t = \tilde{\mathrm{x}}_{t-1} + \frac{\epsilon}{2} \nabla_x \log p(\tilde{\mathrm{x}}_{t-1}) + \sqrt{\epsilon}\mathrm{z}_t$$

  - $\mathrm{z}_t \sim \mathcal{N}(0, I)$
- Score-based generative model
  - Train $s_\theta(\mathrm{x}) \approx \nabla \log p(\mathrm{x})$
  - Draw samples based on Langevin dynamics

Challenges: Low-d Data Manifolds

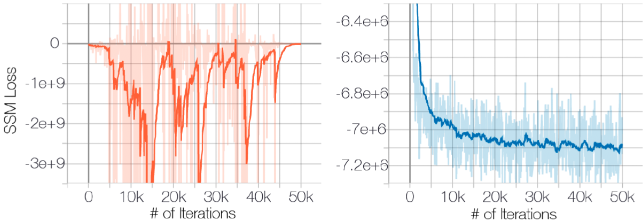


Figure 1: **Left**: Sliced score matching (SSM) loss w.r.t. iterations. No noise is added to data. **Right**: Same but data are perturbed with $\mathcal{N}(0, 0.0001)$.

- Real world data lies in low-d manifolds
- Score $\nabla_x \log p(\mathrm{x})$ is gradient taken in ambient space
- Adding small amount of (ambient) noise seems to help

Challenges: Low Density Regions

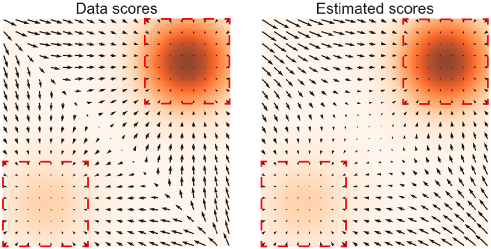


Figure 2: **Left**: $\nabla_{\mathbf{x}} \log p_{\text{data}}(\mathbf{x})$; **Right**: $\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x})$. The data density $p_{\text{data}}(\mathbf{x})$ is encoded using an orange colormap: darker color implies higher density. Red rectangles highlight regions where $\nabla_{\mathbf{x}} \log p_{\text{data}}(\mathbf{x}) \approx \mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x})$.

- Low density regions may not have enough samples
- Cannot estimate $\nabla_x \log p(x)$

Challenges: Mixing of Langevin Dynamics

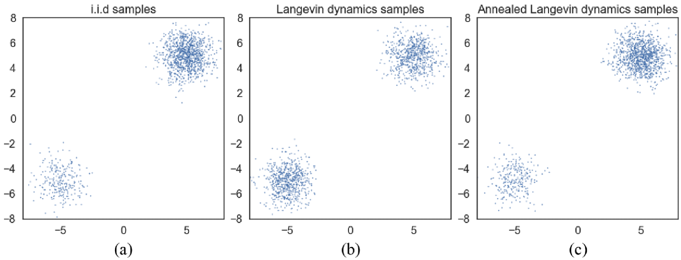


Figure 3: Samples from a mixture of Gaussian with different methods. (a) Exact sampling. (b) Sampling using Langevin dynamics with the exact scores. (c) Sampling using annealed Langevin dynamics with the exact scores. Clearly Langevin dynamics estimate the relative weights between the two modes incorrectly, while annealed Langevin dynamics recover the relative weights faithfully.

- Low density regions separating modes
- Hard to recover relative weights of such disjoint modes
- Modes with disjoint support: $\pi p_1(x) + (1 - \pi)p_2(x)$
  - Score function does not depend on π

Noise Conditional Score Networks

- Choose a decreasing sequence $\{\sigma_i\}_{i=1}^{L}$ with $\frac{\sigma_i}{\sigma_{i+1}} > 1$
  - Noise level σ_1 is big enough to address low-density issues
  - Noise level σ_L is small, does not pertub true distribution by much
- Perturbed distribution

$$q_\sigma(\mathrm{x}) = \int p(t)\mathcal{N}(\mathrm{x}|t, \sigma^2\mathbb{I})dt$$

- Score network jointly estimates score at all perturbations

$$s_\theta(\mathrm{x}, \sigma) \approx \nabla_{\mathrm{x}} \log q_\sigma(\mathrm{x}) \ , \quad \forall \sigma \in \{\sigma_i\}_{i=1}^{L}$$

- $s_\theta(\mathrm{x}, \sigma)$: Noise Conditional Score Network (NCSN)
- Architecture for NCSN: U-nets with dilated/atrous convolutions

Learning NCSNs via Score Matching

- Score matching with denoise using

$$q_\sigma(\tilde{x}|x) = cN(\tilde{x}|x, \sigma^2\mathbb{I}) \quad \Rightarrow \quad \nabla_{\tilde{x}} \log q_\sigma(\tilde{x}|x) = -\frac{(\tilde{x}-x)^2}{\sigma^2}$$

- Denoising score matching objective at any fixed σ

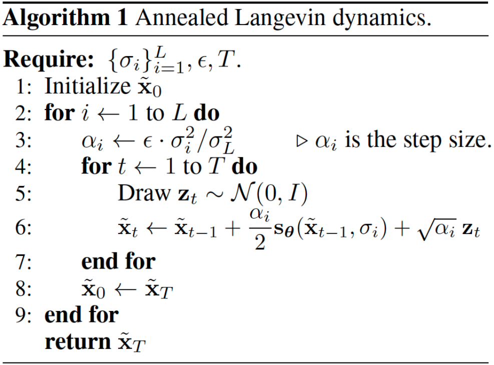
$$\ell(\theta;\sigma) = \frac{1}{2}\mathbb{E}_p(x)\mathbb{E}_{\tilde{x}\sim\mathcal{N}(x,\sigma^2\mathbb{I})}\left[\left\|s_\theta(\tilde{x},\sigma) + \frac{(\tilde{x}-x)^2}{\sigma^2}\right\|^2\right]$$

- Combined objective, for some $\lambda(\sigma_i) > 0$

$$\mathcal{L}(\theta;\{\sigma_i\}_{i=1}^L) = \frac{1}{L}\sum_{i=1}^{L}\lambda(\sigma_i)\ell(\theta;\sigma_i)$$

- Choice of $\lambda(\sigma) \propto \sigma^2$ to make loss invariant to σ
    - Empirically $\|s_\theta(x,\sigma) \propto 1/\sigma$
    - Then, $\sigma^2\ell(\theta;\sigma) = O(1)$

NCSN Inference via Annealed Langevin Dynamics

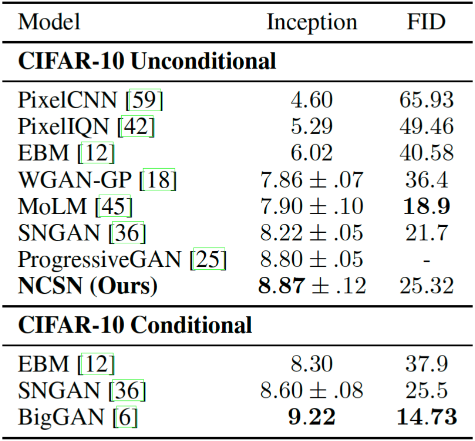


Algorithm 1 Annealed Langevin dynamics.

```
Require: $\{\sigma_i\}_{i=1}^{L}, \epsilon, T$.
Initialize $\tilde{\mathbf{x}}_0$
for $i \leftarrow 1$ to $L$ do
    $\alpha_i \leftarrow \epsilon \cdot \sigma_i^2 / \sigma_L^2$        ▷ $\alpha_i$ is the step size.
    for $t \leftarrow 1$ to $T$ do
        Draw $\mathbf{z}_t \sim \mathcal{N}(0, I)$
        $\tilde{\mathbf{x}}_t \leftarrow \tilde{\mathbf{x}}_{t-1} + \frac{\alpha_i}{2} \mathbf{s}_{\boldsymbol{\theta}}(\tilde{\mathbf{x}}_{t-1}, \sigma_i) + \sqrt{\alpha_i}\, \mathbf{z}_t$
    end for
    $\tilde{\mathbf{x}}_0 \leftarrow \tilde{\mathbf{x}}_T$
end for
return $\tilde{\mathbf{x}}_T$
```

- Run Langevin dynamics at different scales
- Start from noise σ_1, all the way down to σ_L
- Run full dynamics at each level, step decreases for each level

Results: Comparisons



Model	Inception	FID
CIFAR-10 Unconditional		
PixelCNN [59]	4.60	65.93
PixelIQN [42]	5.29	49.46
EBM [12]	6.02	40.58
WGAN-GP [18]	7.86 ± .07	36.4
MoLM [45]	7.90 ± .10	**18.9**
SNGAN [36]	8.22 ± .05	21.7
ProgressiveGAN [25]	8.80 ± .05	-
NCSN (Ours)	**8.87** ± .12	25.32
CIFAR-10 Conditional		
EBM [12]	8.30	37.9
SNGAN [36]	8.60 ± .08	25.5
BigGAN [6]	**9.22**	**14.73**

Table 1: Inception and FID scores for CIFAR-10

Results: Annealed Langevin dynamics

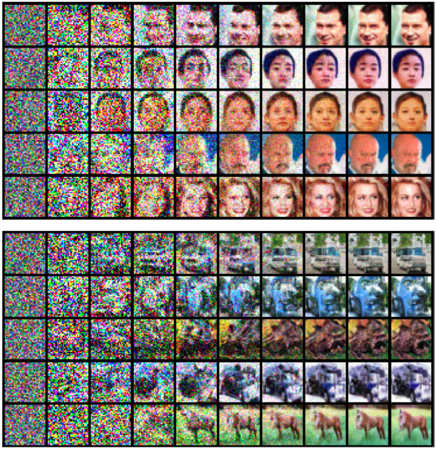
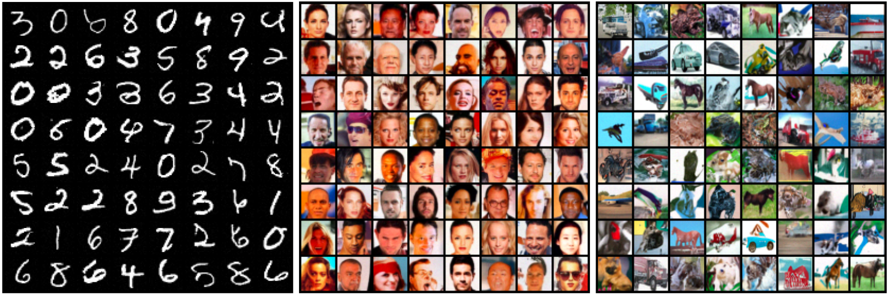


Figure 4: Intermediate samples of annealed Langevin dynamics.

Results: Samples



(a) MNIST (b) CelebA (c) CIFAR-10

Figure 5: Uncurated samples on MNIST, CelebA, and CIFAR-10 datasets.

Results: Inpainting

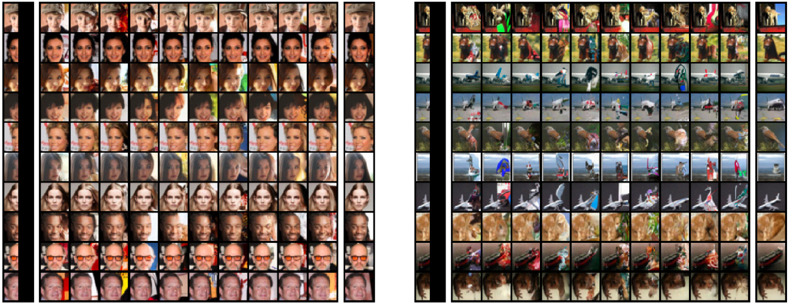


Figure 6: Image inpainting on CelebA (**left**) and CIFAR-10 (**right**). The leftmost column of each figure shows the occluded images, while the rightmost column shows the original images.

References

- A. Hyvarinen. Estimation of non-normalized statistical models by score matching, JMLR, 2005.
- Y. Song, S. Ermon. Generative modeling by estimating gradients of the data distribution, NeurIPS, 2019.