Professur für Machine Learning
Fakultät für Elektro- und Informationstechnik
Technische Universität München

Solving Inverse Problems in Medical Imaging with Score-Based Generative Models

Malek Ben Alaya

January 11, 2023

Linear inverse problem

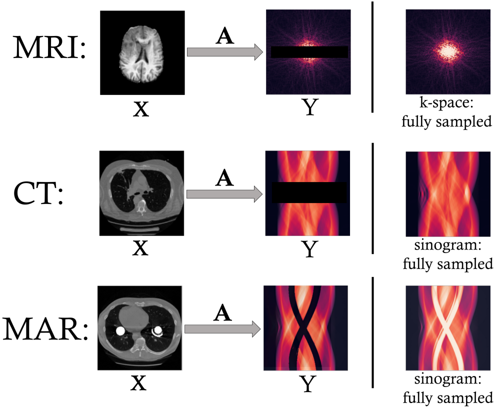
Probelem statement:

Given $y \in \mathbb{R}^m \rightarrow$ recover $x \in \mathbb{R}^n$, where

$$y = Ax + \epsilon \tag{1}$$

$A \in \mathbb{R}^{m \times n}$: Measurement Matrix, $\epsilon \in \mathbb{R}^m$: noise vector

Linear inverse problem: Medical Imaging



Score-based Generative Models: unconditional sampling

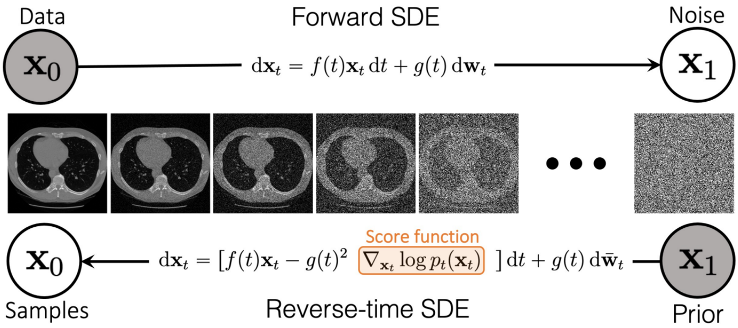


Figure: Generate image samples from prior $p(x)$

Approximate score function with neural network $s_\theta(x_t, t) \approx \nabla_{x_t} \log p_t(x_t)$

Score-based Generative Models: conditional sampling

Inverse problems: sample from conditional $p(x \mid y)$

Procedure:

- condition $\{\mathbf{x}_t\}_{t\in[0,1]}$ on $y \rightarrow$ *yields* $\{\mathbf{x}_t \mid \mathbf{y}\}_{t\in[0,1]}$
- reverse $\{\mathbf{x}_t \mid \mathbf{y}\}_{t\in[0,1]}$ by solving reverse-time SDE:

$$\mathrm{d}\mathbf{x}_t = \left[f(t)\mathbf{x}_t - g(t)^2 \nabla_{\mathbf{x}_t} \log p_t\left(\mathbf{x}_t \mid \mathbf{y}\right)\right] \mathrm{d}t + g(t)\mathrm{d}\overline{\mathbf{w}}_t, \quad t \in [0, 1]$$

- for $t = 0$: $(x_0 \mid y)$ is (approximate) sample from $p(x \mid y)$

Score-based Generative Models: conditional sampling

Problem: computing $\nabla_{\mathbf{x}_t} \log p_t\left(\mathbf{x}_t \mid \mathbf{y}\right)$ in

$$\mathrm{d}\mathbf{x}_t = \left[f(t)\mathbf{x}_t - g(t)^2 \nabla_{\mathbf{x}_t} \log p_t\left(\mathbf{x}_t \mid \mathbf{y}\right)\right] \mathrm{d}t + g(t)\mathrm{d}\overline{\mathbf{w}}_t, \quad t \in [0, 1] \tag{2}$$

Existing solutions :

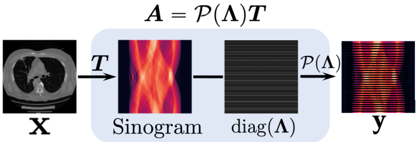
- estimate term with new score model $s_\theta\left(\mathbf{x}_t, \mathbf{y}, t\right) \approx \nabla_{\mathbf{x}_t} \log p_t\left(\mathbf{x}_t \mid \mathbf{y}\right)$: requires $\{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^N \rightarrow$ supervised
- unsupervised: solve (2) using $s_\theta(x_t, t)$ but require SVD of $\boldsymbol{A}$: problematic

Proposed solution:

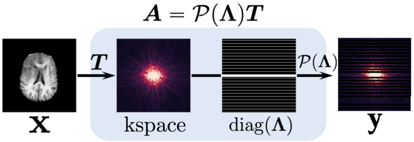
- Solve (2) using $s_\theta(x_t, t)$ and efficient decomposition of $\boldsymbol{A}$ instead of SVD

Solve reverse conditional SDE using $s_\theta(x_t, t)$ and Decomposition of A

- $A = \mathcal{P}(\boldsymbol{\Lambda})\boldsymbol{T}$
- exists if $\mathrm{rank}(A) = m$
- $\boldsymbol{T}$: Radon/Fourier Transformation in CT/MRI
- $\mathrm{diag}(\boldsymbol{\Lambda})$: subsampling mask on sinogram/k-space
- $\mathcal{P}(\boldsymbol{\Lambda})$ subsamples sinogram/k-space into observation y with smaller size



(a) Linear Measurement in CT



(b) Linear Measurement in MRI

Solve reverse conditional SDE using $s_\theta(x_t, t)$: Incorporate observation y into unconditional sampling process

Procedure:

- Sampling from $\{\mathbf{x}_t \mid \mathbf{y}\}_{t\in[0,1]}$ complicated
- Define $\{\mathbf{y}_t\}_{t\in[0,1]}$, where $\mathbf{y}_t = \boldsymbol{A}\mathbf{x}_t + \alpha(t)\boldsymbol{\epsilon} = ... = \alpha(t)\mathbf{y} + \beta(t)\boldsymbol{A}\mathbf{z}$
- Sampling from $\{\mathbf{y}_t \mid \mathbf{y}\}_{t\in[0,1]}$ easy
- Make samples of $\{\mathbf{x}_t\}_{t\in[0,1]}$ consistent with $\{\mathbf{y}_t \mid \mathbf{y}\}_{t\in[0,1]}$
  $\rightarrow$ ensure $\boldsymbol{A}\mathbf{x}_t \approx \mathbf{y}_t$ at each sampling step

Incorporating observation y into unconditional sampling process: Ensure $\{\mathbf{x}_t\}_{t\in[0,1]}$ consistent with $\{\mathbf{y}_t \mid \mathbf{y}\}_{t\in[0,1]}$

Iterative unconditional sampler:

$$\hat{\mathbf{x}}_{t_{i-1}} = \boldsymbol{h}\left(\hat{\mathbf{x}}_{t_i}, \mathbf{z}_i, s_{\theta^*}\left(\hat{\mathbf{x}}_{t_i}, t_i\right)\right), \quad i = N, N-1, \cdots, 1,$$

where $\hat{\mathbf{x}}_{t_N} \sim \pi(\mathbf{x}), \mathbf{z}_i \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$.

Incorporate observation y into unconditional sampling process: Ensure $\{\mathbf{x}_t\}_{t\in[0,1]}$ consistent with $\{\mathbf{y}_t \mid \mathbf{y}\}_{t\in[0,1]}$

Iterative conditional sampler:

$$\text{intermediate sample } \hat{\mathbf{x}}'_{t_i} = \boldsymbol{k}\left(\hat{\mathbf{x}}_{t_i}, \hat{\mathbf{y}}_{t_i}, \lambda\right) \text{ ensure consistency: } A\hat{\mathbf{x}}'_{t_i} \approx \hat{\mathbf{y}}_{t_i}$$

$$\hat{\mathbf{x}}_{t_{i-1}} = \boldsymbol{h}\left(\hat{\mathbf{x}}'_{t_i}, \mathbf{z}_i, \boldsymbol{s}_{\theta}\left(\hat{\mathbf{x}}_{t_i}, t_i\right)\right), \quad i = N, N-1, \cdots, 1,$$

where $\hat{\mathbf{y}}_{t_i} \sim p_{t_i}\left(\mathbf{y}_{t_i} \mid \mathbf{y}\right)$

$0 \leqslant \lambda \leqslant 1$: balances between $\hat{\mathbf{x}}'_{t_i} \approx \hat{\mathbf{x}}_{t_i}$ and $A\hat{\mathbf{x}}'_{t_i} \approx \hat{\mathbf{y}}_{t_i}$:

- $\lambda = 0 \rightarrow \hat{\mathbf{x}}'_{t_i} = \boldsymbol{k}\left(\hat{\mathbf{x}}_{t_i}, \hat{\mathbf{y}}_{t_i}, 0\right) = \hat{\mathbf{x}}_{t_i} \rightarrow$ unconditional sampling
- $\lambda = 1 \rightarrow \hat{\mathbf{x}}'_{t_i} = \boldsymbol{k}\left(\hat{\mathbf{x}}_{t_i}, \hat{\mathbf{y}}_{t_i}, 1\right)$ satisfies $\boldsymbol{A}\hat{\mathbf{x}}'_{t_i} = \hat{\mathbf{y}}_{t_i}$

$\boldsymbol{k}\left(\hat{\mathbf{x}}_{t_i}, \hat{\mathbf{y}}_{t_i}, \lambda\right) = \boldsymbol{T}^{-1}\left[\lambda\boldsymbol{\Lambda}\mathcal{P}^{-1}(\boldsymbol{\Lambda})\hat{\mathbf{y}}_{t_i} + (1-\lambda)\boldsymbol{\Lambda}\boldsymbol{T}\hat{\mathbf{x}}_{t_i} + (\boldsymbol{I}-\boldsymbol{\Lambda})\boldsymbol{T}\hat{\mathbf{x}}_{t_i}\right]$ use A decomposition

Incorporate observation y into unconditional sampling process

Euler-Maruyama sampler:

Algorithm 1 Unconditional sampling

Require: N

$\hat{\mathbf{x}}_1 \sim \pi(\mathbf{x}), \Delta t \leftarrow \frac{1}{N}$

for $i = N - 1$ **to** 0 **do**

$\quad t \leftarrow \frac{i+1}{N}$

$\quad \hat{\mathbf{x}}_{t-\Delta t} \leftarrow \hat{\mathbf{x}}_t - f(t)\hat{\mathbf{x}}_t \Delta t$

$\quad \hat{\mathbf{x}}_{t-\Delta t} \leftarrow \hat{\mathbf{x}}_{t-\Delta t} + g(t)^2 \boldsymbol{s}_{\boldsymbol{\theta}*}(\hat{\mathbf{x}}_t, t)\Delta t$

$\quad \mathbf{z} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$

$\quad \hat{\mathbf{x}}_{t-\Delta t} \leftarrow \hat{\mathbf{x}}_{t-\Delta t} + g(t)\sqrt{\Delta t}\, \mathbf{z}$

return $\hat{\mathbf{x}}_0$

Algorithm 2 Inverse problem solving

Require: $N, \mathbf{y}, \lambda$

$\hat{\mathbf{x}}_1 \sim \pi(\mathbf{x}), \Delta t \leftarrow \frac{1}{N}$

for $i = N - 1$ **to** 0 **do**

$\quad t \leftarrow \frac{i+1}{N}$

$\quad \hat{\mathbf{y}}_t \sim p_{0t}(\mathbf{y}_t \mid \mathbf{y})$

$\quad \hat{\mathbf{x}}_t \leftarrow \boldsymbol{T}^{-1}[\lambda \boldsymbol{\Lambda} \mathcal{P}^{-1}(\boldsymbol{\Lambda})\hat{\mathbf{y}}_t + (1-\lambda)\boldsymbol{\Lambda}\boldsymbol{T}\hat{\mathbf{x}}_t + (\boldsymbol{I} - \boldsymbol{\Lambda})\boldsymbol{T}\hat{\mathbf{x}}_t]$

$\quad \hat{\mathbf{x}}_{t-\Delta t} \leftarrow \hat{\mathbf{x}}_t - f(t)\hat{\mathbf{x}}_t \Delta t$

$\quad \hat{\mathbf{x}}_{t-\Delta t} \leftarrow \hat{\mathbf{x}}_{t-\Delta t} + g(t)^2 \boldsymbol{s}_{\boldsymbol{\theta}*}(\hat{\mathbf{x}}_t, t)\Delta t$

$\quad \mathbf{z} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$

$\quad \hat{\mathbf{x}}_{t-\Delta t} \leftarrow \hat{\mathbf{x}}_{t-\Delta t} + g(t)\sqrt{\Delta t}\, \mathbf{z}$

return $\hat{\mathbf{x}}_0$

Experiments and results

Datasets:

- CT/ MAR:
  - LIDC (Lungs): **Training/ Testing:** 130K/ 1K images
  - LDCT (Head, Chest): **Training/ Testing:** 47K/ 1K images
- MRI:
  - BraTS (Brain): **Training/ Testing:** 297K/ 1K images

Evaluation: PSNR and SSIM

Experiments and results: CT Reconstruction

	Method	Projections (test time)	LIDC 320 × 320		LDCT 512 × 512	
			PSNR↑	SSIM↑	PSNR↑	SSIM↑
learning-free	FBP	23	10.18±1.38	0.230±0.072	10.11±1.19	0.302±0.078
	FISTA-TV	23	20.08±4.89	0.799±0.061	21.88±4.42	0.850±0.067
supervised	cGAN	23	19.83±3.07	0.479±0.103	19.90±2.52	0.545±0.065
	Neumann	23	17.18±3.79	0.454±0.128	18.83±3.29	0.525±0.073
	SIN-4c-PRN	23	30.48±3.99	0.895±0.047	34.82±3.55	0.877±0.116
	Ours	10	29.52±2.63	0.823±0.061	28.96±4.41	0.849±0.086
		20	34.40±2.66	0.895±0.048	36.80±4.50	0.936±0.058
		23	**35.24**±2.71	**0.905**±0.046	**37.41**±4.62	**0.941**±0.057

Figure: Results for sparse-view CT reconstruction. Supervised methods are trained with 23 projection angles.

- ’Ours’ better than supervised even when given less measurements

Experiments and results: CT Reconstruction

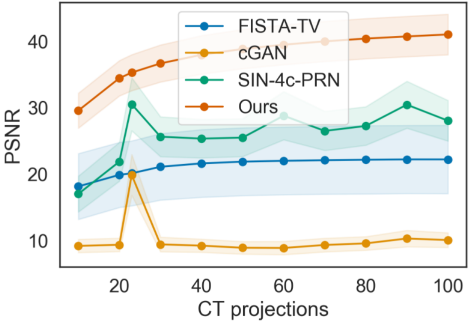


Figure: CT on LIDC

- Recall: supervised trained on 23 projections
- 'Ours' generalizes better than supervised.

Experiments and results: CT Reconstruction

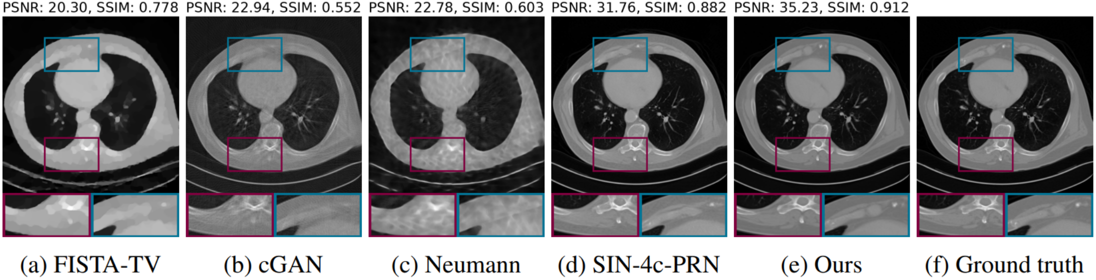


Figure: Visual comparison of sparse-view CT reconstruction quality

Experiments and results: MAR

	Method	PSNR↑	SSIM↑
learning-free	LI	26.30±2.62	0.910±0.028
supervised	cGANMAR	27.27±1.96	0.927±0.060
	SNMAR	27.28±1.43	0.937±0.048
	Ours	**32.16**±2.32	**0.939**±0.022

Figure: MAR results on LIDC.

- Supervised methods trained specifically for MAR
- 'Ours' not trained specifically for MAR, still outperforms supervised methods trained for MAR
  $\rightarrow$ generalizes to different measurement process

Experiments and results: MRI Reconstruction

	Method	24× Acceleration		8× Acceleration		4× Acceleration	
		PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑
supervised	Cascade DenseNet	23.39±2.17	0.765±0.042	28.35±2.30	0.845±0.038	30.97±2.33	0.902±0.028
	DuDoRNet	18.46±3.05	0.662±0.093	**37.88±3.03**	**0.985±0.007**	30.53±4.13	0.891±0.071
unsupervised	Score SDE	27.83±2.73	0.849±0.038	35.04±2.11	0.943±0.016	37.55±2.08	0.960±0.013
	Langevin	28.80±3.21	0.873±0.039	36.44±2.28	0.952±0.016	38.76±2.32	0.966±0.012
	Ours	**29.42±3.03**	**0.880±0.035**	37.63±2.70	0.958±0.015	**39.91±2.67**	**0.965±0.013**

Figure: Results for undersampled MRI reconstruction on BraTS. Supervised methods are trained with 8x acceleration.

Experiments and results: MRI Reconstruction

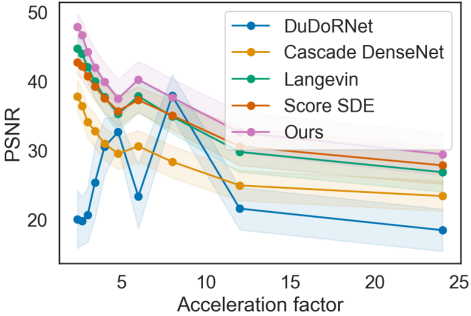


Figure: MRI on BraTS

- Unsupervised generalizes better, 'Ours' even better

Experiments and results: Effect of more advanced Samplers

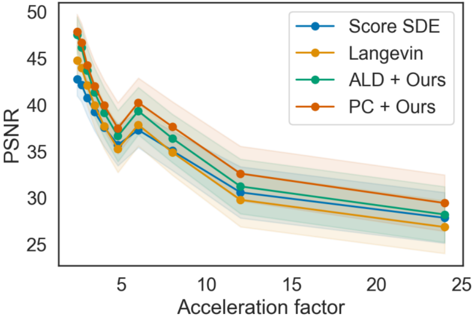


Figure: MRI on BraTS

- 'Ours' uniformly better than existing score-based methods
- Better samplers $\rightarrow$ better performance

Conclusion

Pros:

- Better than existing methods in CT, MAR
- Better generalization to different number of measurements in CT and MRI
- Same score model solves CT and MAR $\rightarrow$ generalization to different measurement process

Cons:

- Not always the best in MRI