FLATIRON INSTITUTE

Dusting off the Cosmic Microwave Background with Diffusion Models

Bruno Régaldo-Saint Blancard
Research Fellow @ Flatiron Institute, New York

Physics in the AI era @ University of Pisa
September 25, 2024

With: D. Heurtel-Depeiges, R. Ohana, B. Burkhart, C. Margossian

Credit: ESO/P. Horálek

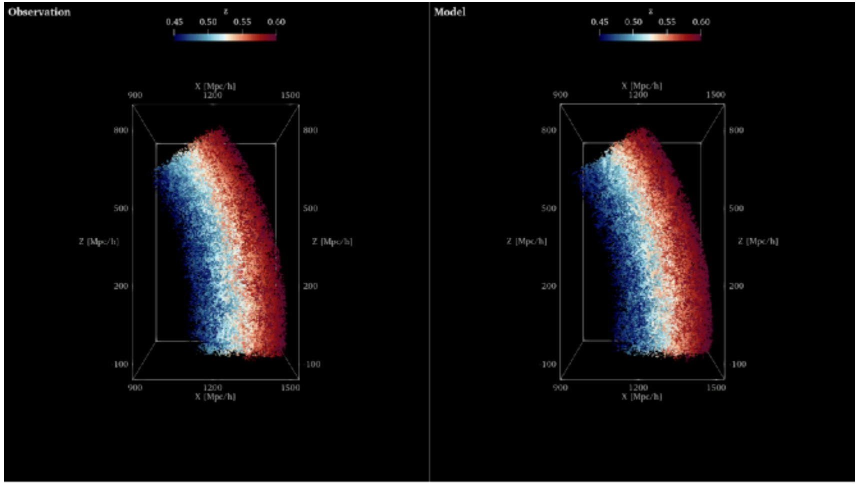
Also happy to chat about…

SIMBIG

Simulation-Based Inference

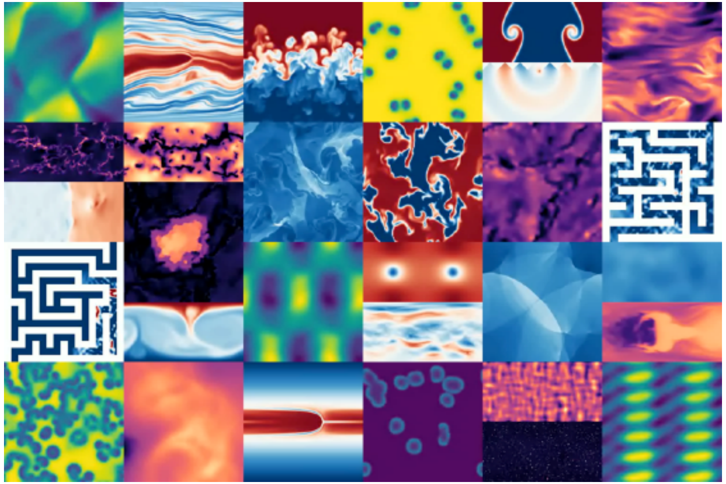
x

Galaxy Clustering



Polymathic

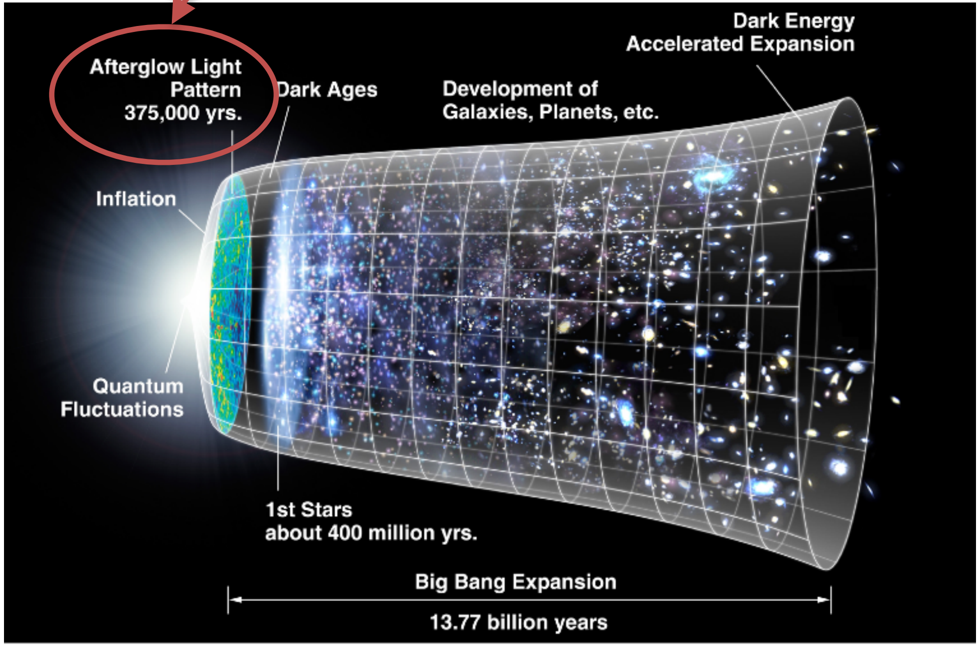
Foundation Models for Science



History of the Universe: an Inference Problem

Cosmic Microwave Background (CMB)

→ Are there *B*-modes in the CMB?



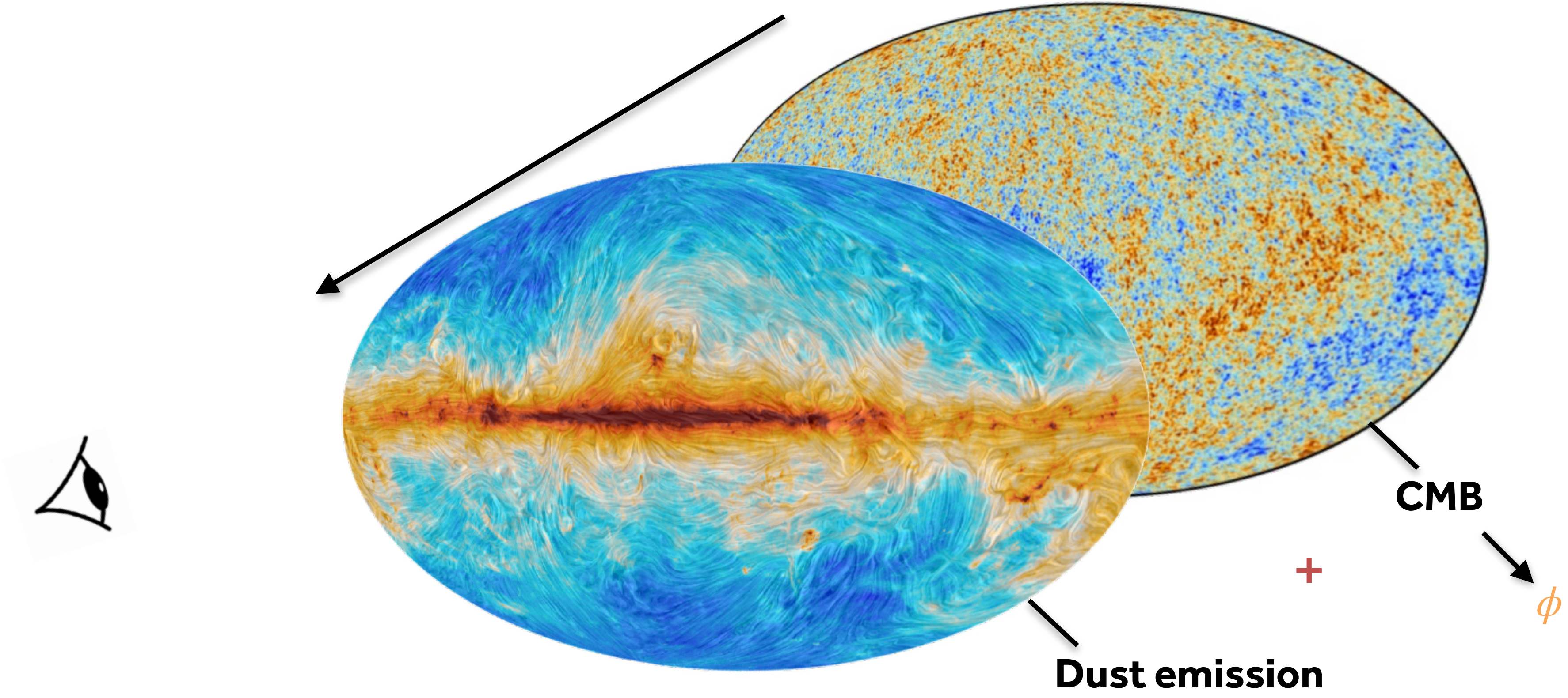
Surprisingly well described by the simple ΛCDM model

But the values of the parameters of this model can change a lot the story...

$$\phi = (\Omega_m, \Omega_b, h, n_s, \sigma_8, \ldots)$$

- Ω_m: Matter density fraction
- Ω_b: Baryon density fraction
- h: Reduced Hubble constant
- n_s: Spectral index of adiabatic perturbations
- σ_8: Mass fluctuation amplitude

Dust = **Foreground** to the Cosmic Microwave Background



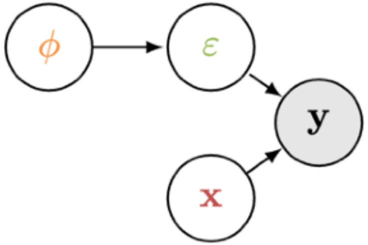
Statistical Challenges

1) Need for a **refined model of the dust emission**.

2) Need for a reliable **component separation method**.

3) Need to combine the above for **cosmological inference**.

Formalization of the Problem



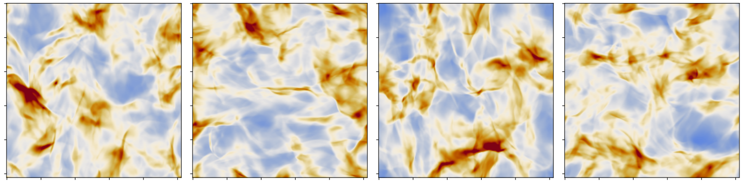
Dust CMB

- We observe $\boxed{\mathbf{y} = \mathbf{x} + \varepsilon}$
- The **CMB is Gaussian** (!) $\varepsilon \sim \mathcal{N}(0, \sigma^2 \Sigma_\phi)$ (to a very good approximation)

  cosmological parameters (typically 6 params.)

- We assume that we have a "prior model" of dust taking the form of a **set of examples:** $\mathcal{T} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$

  e.g. **simulations** →



Burkhart et al. (2020)

- Two goals:

  1) Component Separation → Sampling $p(\varepsilon, \mathbf{x} \mid \mathbf{y}, \phi)$ (but $p(\mathbf{x} \mid \mathbf{y}, \phi)$ is enough!)

  2) Cosmological Inference → Sampling $p(\phi \mid \mathbf{y})$

Reminder...

Dust → $\mathbf{x}$, CMB → ε

$$\mathbf{y} = \mathbf{x} + \varepsilon$$

ϕ **is fixed here!**

1. Component Separation with Diffusion Models

D. Heurtel-Depeiges, B. Burkhart, R. Ohana, **BRSB**, MLPS NeurIPS Workshop (2023) (Oral)

astro-ph.CO arXiv:2310.16285

Sampling of $p(\mathbf{x} \,|\, \mathbf{y}, \phi) \propto p(\mathbf{y} \,|\, \mathbf{x}, \phi) p(\mathbf{x})$

likelihood $\sim \mathcal{N}(\mathbf{x}, \sigma^2 \Sigma_\phi)$

dust prior
→ **needs to be modeled**

Diffusion Models for Generative Modeling

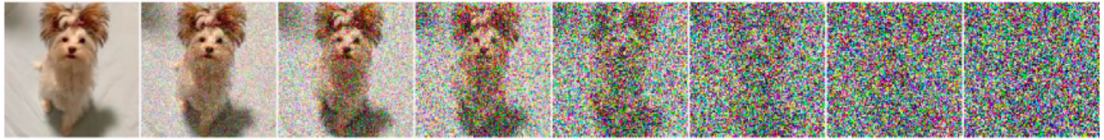
Sohl-Dickstein et al. (2015), Ho et al. (2020), Song et al. (2021)

Forward SDE: $$\mathrm{d}\mathbf{z}_t = -\frac{1}{2}\beta(t)\mathbf{z}_t\,\mathrm{d}t + \sqrt{\beta(t)}\,\mathrm{d}\mathbf{w}_t$$

Easy

$t = 0$

$\mathbf{z}_0 \sim p_{\text{target}}$



$t = 1$

$\mathbf{z}_1 \sim \mathcal{N}(0, I_d)$

Hard

Backward SDE: $$\mathrm{d}\mathbf{z}_t = \left[-\frac{1}{2}\beta(t)\mathbf{z}_t - \beta(t)\underbrace{\nabla_{\mathbf{z}_t}\log p_t(\mathbf{z}_t)}_{\text{Score function}}\right]\mathrm{d}t + \sqrt{\beta(t)}\,\mathrm{d}\overline{\mathbf{w}}_t$$

Idea 1: Diffusion Models as a Mapping $p_{\text{dust}} \leftrightarrow p_{\text{CMB}}$

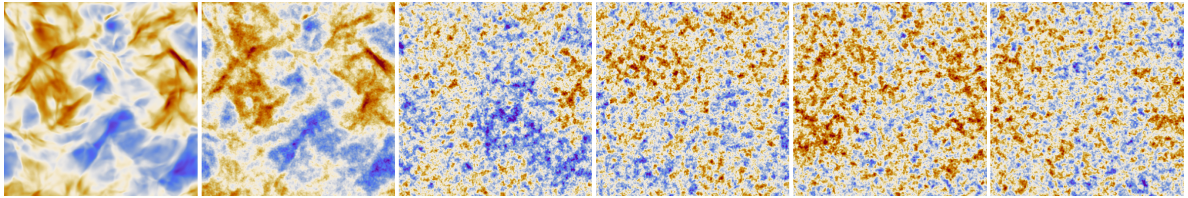
Forward SDE: $$\mathrm{d}\mathbf{z}_t = -\frac{1}{2}\beta(t)\mathbf{z}_t\,\mathrm{d}t + \sqrt{\beta(t)}\boldsymbol{\Sigma}_\phi^{\frac{1}{2}}\,\mathrm{d}\mathbf{w}_t$$

Easy

$t = 0$

$\mathbf{z}_0 \sim p_{\text{dust}}$

$\mathscr{T} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$



$t = 1$

$\mathbf{z}_1 \sim p_{\text{CMB}}$

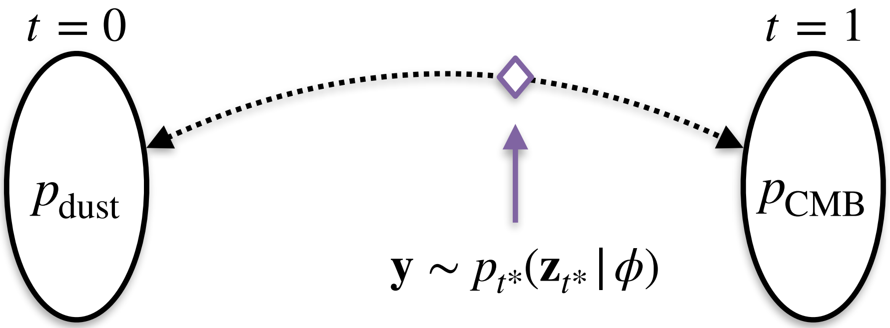
Hard

Backward SDE: $$\mathrm{d}\mathbf{z}_t = \left[-\frac{1}{2}\beta(t)\mathbf{z}_t - \beta(t)\boldsymbol{\Sigma}_\phi \nabla_{\mathbf{z}_t}\log p_t(\mathbf{z}_t)\right]\mathrm{d}t + \sqrt{\beta(t)}\boldsymbol{\Sigma}_\phi^{\frac{1}{2}}\,\mathrm{d}\overline{\mathbf{w}}_t$$

Score function

Idea 2: Component Separation = Solving Backward SDE

Component Separation → Sampling $p(\mathbf{x} \mid \mathbf{y}, \phi)$



Sampling $p(\mathbf{x} \mid \mathbf{y}, \phi) \leftrightarrow$ Solving the Backward SDE starting from $\mathbf{y}$

Anderson (1982)

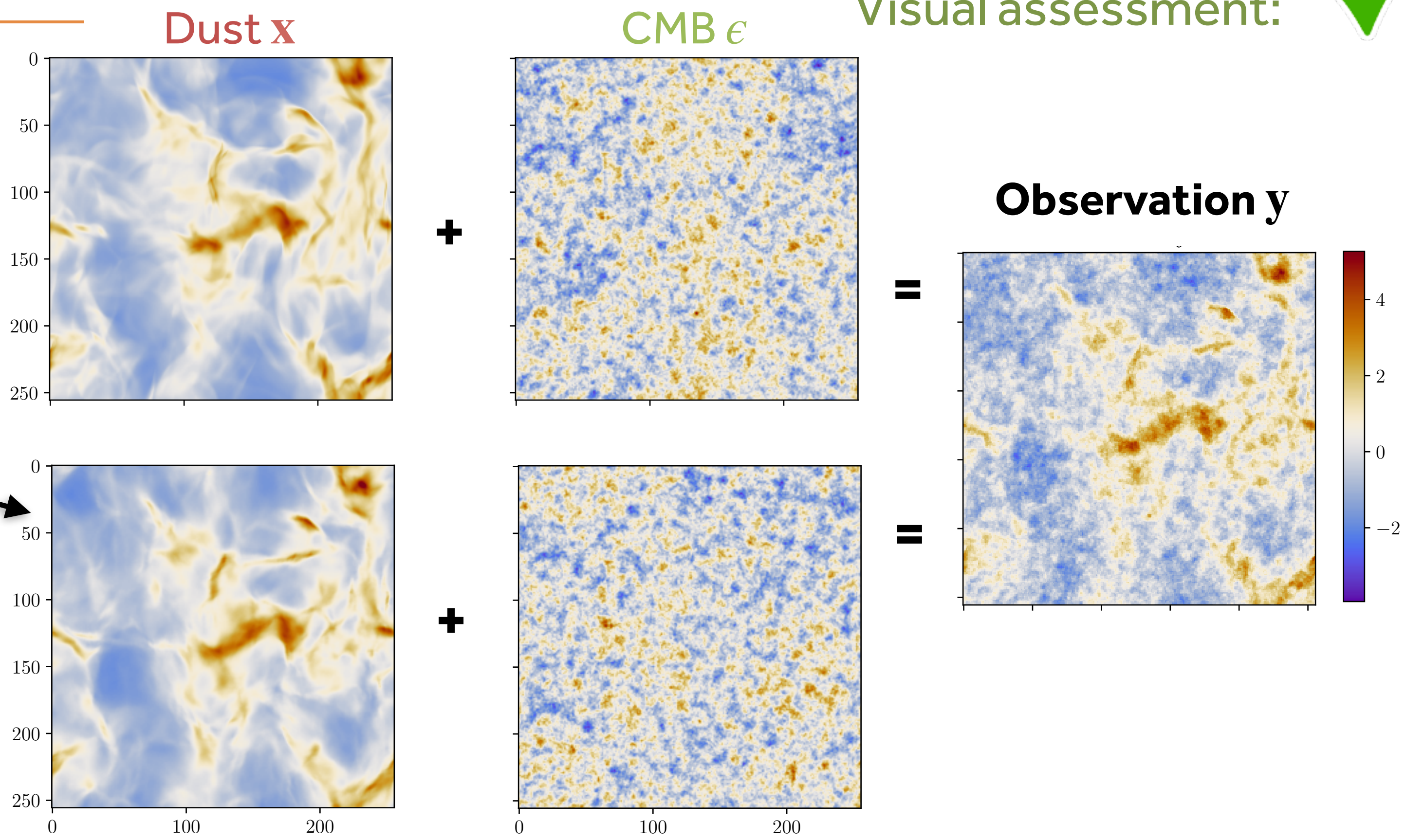
Component Separation ?

Visual assessment: ✓

True

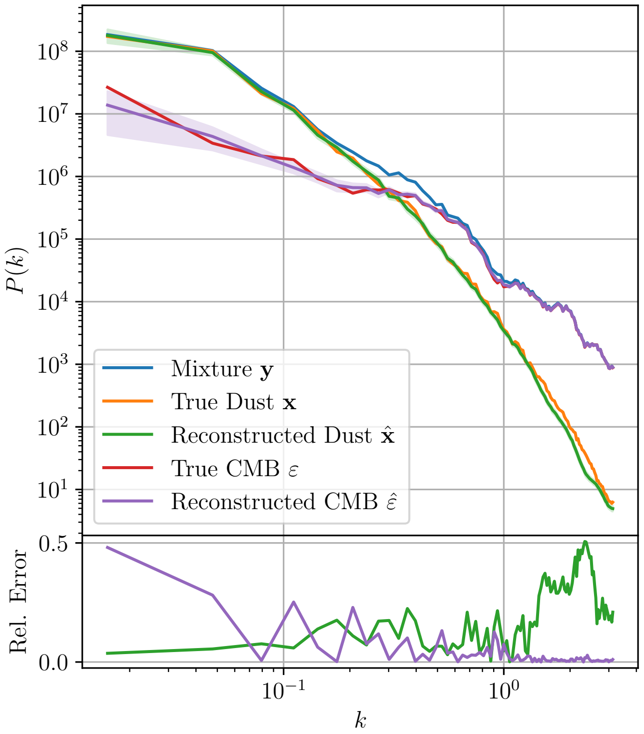
Sample from
$p(\mathbf{x} \,|\, \mathbf{y}, \phi)$

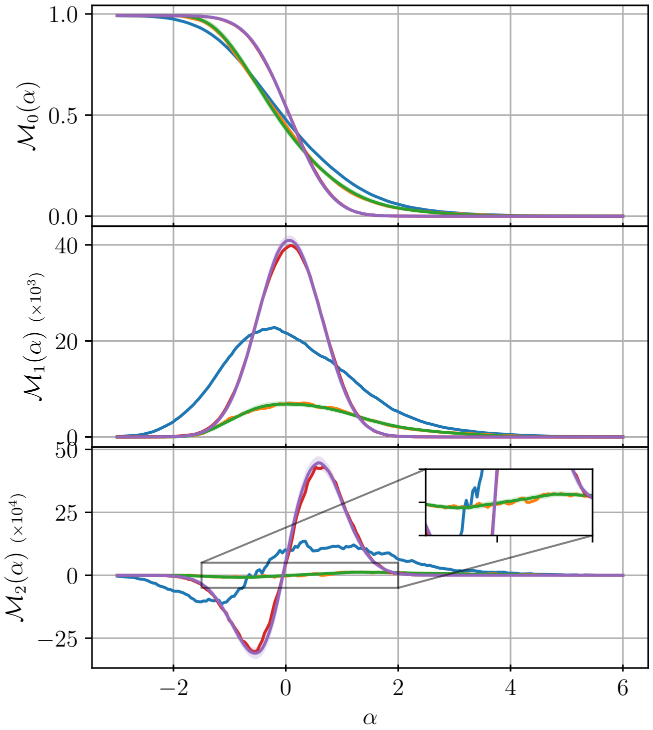
Reconstructed



Statistics of the Reconstruction

Quantitative assessment:





Reminder...

Dust → $\mathbf{x}$, CMB → ε

$$\mathbf{y} = \mathbf{x} + \varepsilon \quad \text{with} \quad \varepsilon \sim \mathcal{N}(0, \sigma^2 \mathbf{\Sigma}_\phi)$$

2. Cosmological Inference

D. Heurtel-Depeiges, C. Margossian, R. Ohana, **BRSB**, ICML (2024)

stat.ML arXiv:2402.19455

rubenohana/Gibbs-Diffusion

Sampling of $p(\phi \mid \mathbf{y})$

Cosmological Inference: Recipe

Goal: Sampling of $p(\phi \,|\, \mathbf{y})$

Ingredients:

1) Dust prior $p_{\text{dust}}(\mathbf{x})$ (learned through examples)
2) Prior on the cosmological parameters $p(\phi)$
3) A diffusion model enabling the sampling of $p(\mathbf{x} \,|\, \mathbf{y}, \phi)$

Recipe: **Gibbs sampling** of $p(\phi, \mathbf{x} \,|\, \mathbf{y})$

Cosmological Inference: Recipe

<u>Recipe</u>: Gibbs sampling of $p(\phi, \mathbf{x} \,|\, \mathbf{y})$ $\rightarrow$ **“Gibbs Diffusion” (GDiff)**

Create a Markov Chain $(\mathbf{x}_k, \phi_k)_{0 \leq k \leq N}$ as follows:

$\phi_0 \sim p_{\text{init}}(\phi)$

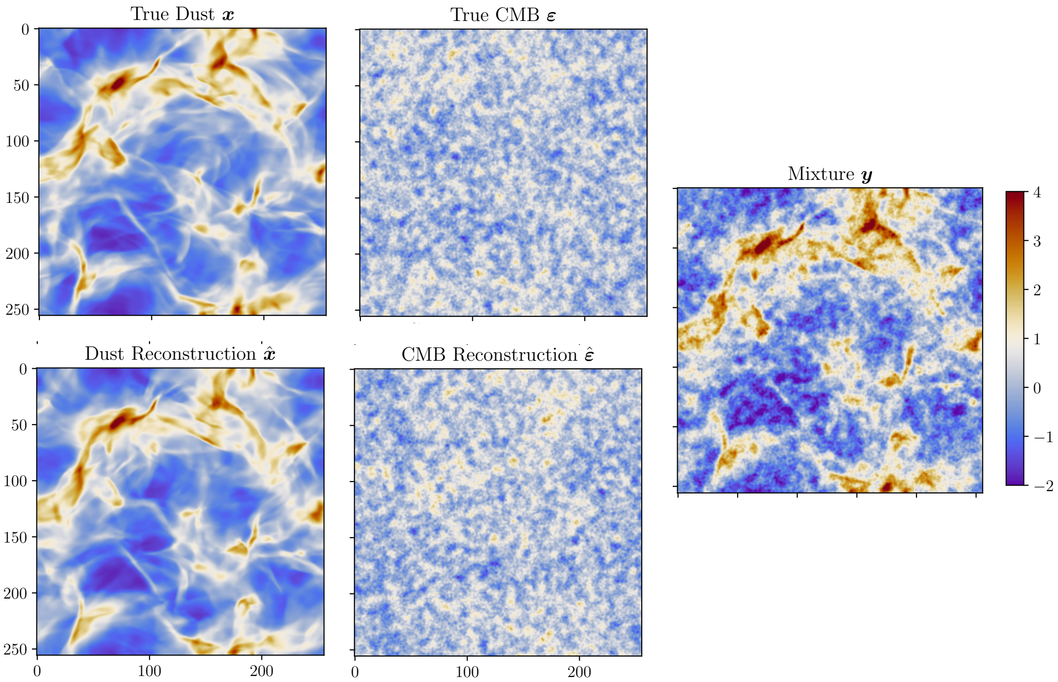
$\mathbf{x}_0 \sim p(\mathbf{x} \,|\, \mathbf{y}, \phi_0)$ ← Diffusion Model

for $k \in \{1, \ldots N\}$:

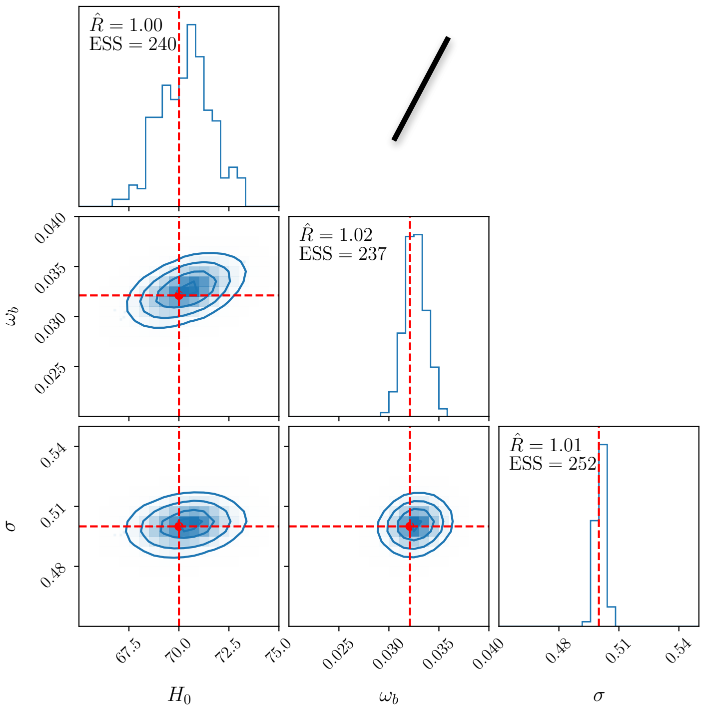
- $\phi_k \sim p(\phi \,|\, \mathbf{y}, \mathbf{x}_{k-1})$ ← Hamiltonian Monte Carlo
- $\mathbf{x}_k \sim p(\mathbf{x} \,|\, \mathbf{y}, \phi_k)$ ← Diffusion Model

Inference Example

$\phi = (H_0, \omega_b, \sigma)$



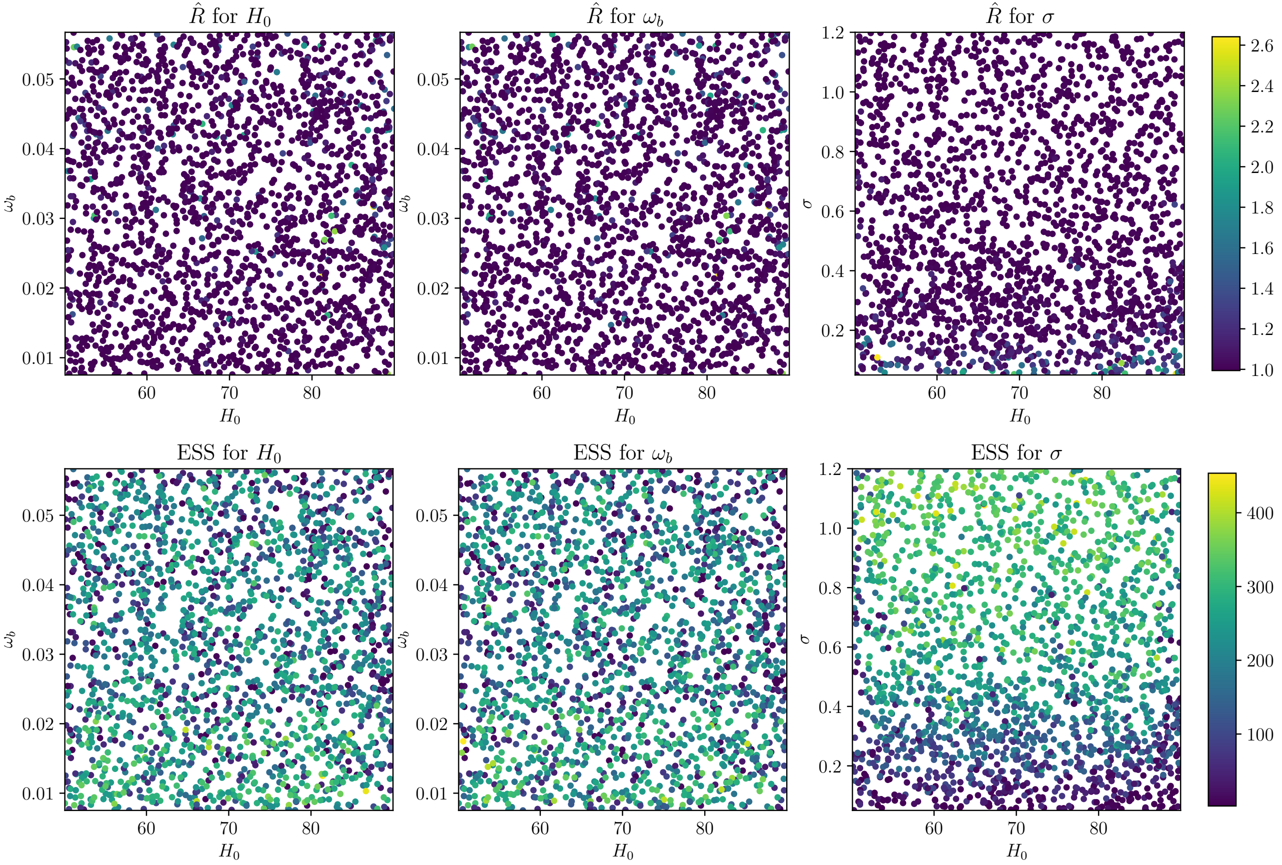
$p(\phi \mid \mathbf{y})$



Validation of the Bayesian Computation

$\hat{R}$ **diagnostic**
for convergence

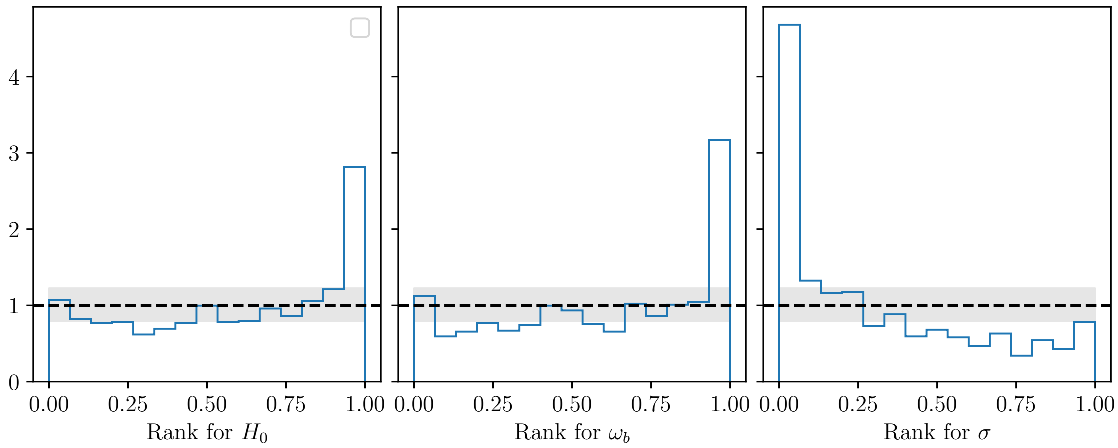
ESS diagnostic
for sampling efficiency



Validation of the Bayesian Computation

Simulation-Based Calibration
diagnostic for posterior estimation accuracy

Talts et al. (2018)



- Ranks for H_0 and ω_b are $\sim$ compatible with uniform distribution
- Ranks for σ indicates some bias introduced by the diffusion model

$$\mathbf{y} = \mathbf{x} + \varepsilon \quad \text{with} \quad \varepsilon \sim \mathcal{N}(0, \sigma^2 \boldsymbol{\Sigma}_{\phi})$$

3. “Gibbs Diffusion”: a Generic Method for Blind Denoising

D. Heurtel-Depeiges, C. Margossian, R. Ohana, **BRSB**, ICML (2024)

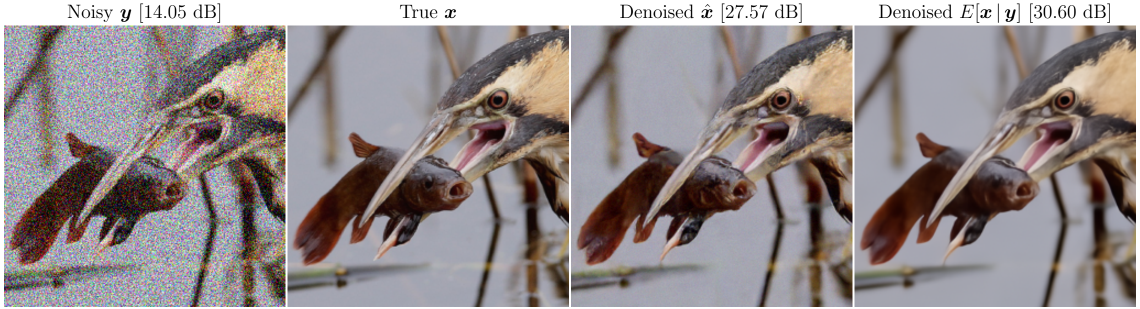
stat.ML arXiv:2402.19455

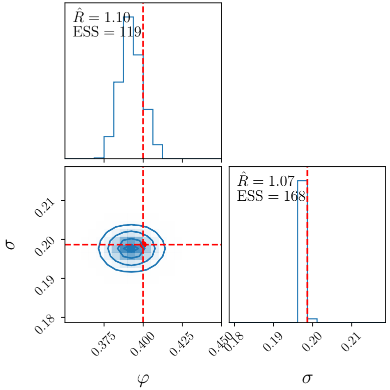
rubenohana/Gibbs-Diffusion

Blind Denoising = Sampling of $p(\mathbf{x}, \phi \mid \mathbf{y})$

Blind Denoising on Natural Images

- $\mathbf{y} = \mathbf{x} + \varepsilon$ with ε a colored noise with unknown exponent (power spectrum is $S_{\varphi}(\mathbf{k}) \propto k^{\varphi}$)
- Diffusion model conditioned on φ trained on ImageNet





Denoising Benchmark

✦ Peak signal-to-noise ratio

Dataset	Noise Level σ	$\varphi = -1 \rightarrow$ Pink noise				$\varphi = 0 \rightarrow$ White noise				$\varphi = 1 \rightarrow$ Blue noise			
		BM3D	DnCNN	GDiff $\hat{\boldsymbol{x}}$	GDiff $\mathbb{E}[\boldsymbol{x} \mid \boldsymbol{y}]$	BM3D	DnCNN	GDiff $\hat{\boldsymbol{x}}$	GDiff $\mathbb{E}[\boldsymbol{x} \mid \boldsymbol{y}]$	BM3D	DnCNN	GDiff $\hat{\boldsymbol{x}}$	GDiff $\mathbb{E}[\boldsymbol{x} \mid \boldsymbol{y}]$
ImageNet	0.06	$31.0_{\pm 0.2}$	$30.2_{\pm 0.2}$	$29.3_{\pm 0.3}$	$\mathbf{32.2}_{\pm 0.3}$	$33.7_{\pm 0.3}$	$33.4_{\pm 0.3}$	$31.5_{\pm 0.3}$	$\mathbf{34.4}_{\pm 0.3}$	$34.7_{\pm 0.4}$	$33.8_{\pm 0.4}$	$32.3_{\pm 0.4}$	$\mathbf{35.3}_{\pm 0.4}$
	0.1	$27.8_{\pm 0.2}$	$26.8_{\pm 0.1}$	$26.7_{\pm 0.2}$	$\mathbf{29.4}_{\pm 0.2}$	$31.8_{\pm 0.3}$	$31.8_{\pm 0.4}$	$29.7_{\pm 0.4}$	$\mathbf{32.7}_{\pm 0.4}$	$32.1_{\pm 0.4}$	$31.5_{\pm 0.3}$	$29.9_{\pm 0.4}$	$\mathbf{32.9}_{\pm 0.3}$
	0.2	$23.5_{\pm 0.2}$	$21.7_{\pm 0.1}$	$23.0_{\pm 0.3}$	$\mathbf{25.7}_{\pm 0.3}$	$28.1_{\pm 0.4}$	$28.4_{\pm 0.4}$	$26.5_{\pm 0.4}$	$\mathbf{29.3}_{\pm 0.4}$	$29.5_{\pm 0.4}$	$28.6_{\pm 0.4}$	$27.6_{\pm 0.4}$	$\mathbf{30.5}_{\pm 0.4}$
CBSD68	0.06	$31.2_{\pm 0.2}$	$30.6_{\pm 0.1}$	$29.2_{\pm 0.2}$	$\mathbf{32.2}_{\pm 0.2}$	$33.8_{\pm 0.3}$	$34.2_{\pm 0.3}$	$31.2_{\pm 0.3}$	$\mathbf{34.4}_{\pm 0.3}$	$35.0_{\pm 0.3}$	$34.8_{\pm 0.3}$	$32.2_{\pm 0.3}$	$\mathbf{35.5}_{\pm 0.3}$
	0.1	$27.9_{\pm 0.2}$	$26.9_{\pm 0.1}$	$26.2_{\pm 0.3}$	$\mathbf{29.1}_{\pm 0.3}$	$31.3_{\pm 0.3}$	$31.7_{\pm 0.3}$	$28.6_{\pm 0.3}$	$\mathbf{31.8}_{\pm 0.3}$	$33.0_{\pm 0.3}$	$32.7_{\pm 0.3}$	$30.6_{\pm 0.4}$	$\mathbf{33.8}_{\pm 0.4}$
	0.2	$23.5_{\pm 0.2}$	$21.7_{\pm 0.1}$	$23.0_{\pm 0.3}$	$\mathbf{25.6}_{\pm 0.2}$	$27.8_{\pm 0.3}$	$28.2_{\pm 0.3}$	$25.4_{\pm 0.3}$	$\mathbf{28.5}_{\pm 0.3}$	$29.6_{\pm 0.3}$	$28.9_{\pm 0.2}$	$27.4_{\pm 0.3}$	$\mathbf{30.6}_{\pm 0.3}$

✦ Structural similarity

Dataset	Noise Level σ	$\varphi = -1 \rightarrow$ Pink noise				$\varphi = 0 \rightarrow$ White noise				$\varphi = 1 \rightarrow$ Blue noise			
		BM3D	DnCNN	GDiff $\hat{\boldsymbol{x}}$	GDiff $\mathbb{E}[\boldsymbol{x} \mid \boldsymbol{y}]$	BM3D	DnCNN	GDiff $\hat{\boldsymbol{x}}$	GDiff $\mathbb{E}[\boldsymbol{x} \mid \boldsymbol{y}]$	BM3D	DnCNN	GDiff $\hat{\boldsymbol{x}}$	GDiff $\mathbb{E}[\boldsymbol{x} \mid \boldsymbol{y}]$
ImageNet	0.06	$0.90_{\pm 0.01}$	$0.88_{\pm 0.00}$	$0.86_{\pm 0.01}$	$\mathbf{0.92}_{\pm 0.00}$	$0.92_{\pm 0.00}$	$0.92_{\pm 0.00}$	$0.88_{\pm 0.01}$	$\mathbf{0.93}_{\pm 0.00}$	$0.94_{\pm 0.00}$	$0.92_{\pm 0.00}$	$0.90_{\pm 0.00}$	$\mathbf{0.95}_{\pm 0.00}$
	0.1	$0.81_{\pm 0.01}$	$0.76_{\pm 0.01}$	$0.77_{\pm 0.01}$	$\mathbf{0.86}_{\pm 0.01}$	$0.90_{\pm 0.01}$	$0.90_{\pm 0.01}$	$0.84_{\pm 0.01}$	$\mathbf{0.91}_{\pm 0.00}$	$0.90_{\pm 0.01}$	$0.88_{\pm 0.01}$	$0.85_{\pm 0.01}$	$\mathbf{0.92}_{\pm 0.00}$
	0.2	$0.63_{\pm 0.01}$	$0.53_{\pm 0.01}$	$0.62_{\pm 0.02}$	$\mathbf{0.74}_{\pm 0.02}$	$0.79_{\pm 0.01}$	$0.80_{\pm 0.01}$	$0.74_{\pm 0.01}$	$\mathbf{0.83}_{\pm 0.01}$	$0.83_{\pm 0.01}$	$0.79_{\pm 0.01}$	$0.78_{\pm 0.01}$	$\mathbf{0.87}_{\pm 0.01}$
CBSD68	0.06	$0.90_{\pm 0.01}$	$0.88_{\pm 0.00}$	$0.84_{\pm 0.01}$	$\mathbf{0.92}_{\pm 0.00}$	$0.93_{\pm 0.00}$	$\mathbf{0.94}_{\pm 0.00}$	$0.88_{\pm 0.00}$	$\mathbf{0.94}_{\pm 0.00}$	$0.94_{\pm 0.00}$	$0.94_{\pm 0.00}$	$0.90_{\pm 0.00}$	$\mathbf{0.95}_{\pm 0.00}$
	0.1	$0.82_{\pm 0.01}$	$0.78_{\pm 0.01}$	$0.75_{\pm 0.01}$	$\mathbf{0.85}_{\pm 0.01}$	$0.89_{\pm 0.00}$	$\mathbf{0.90}_{\pm 0.00}$	$0.81_{\pm 0.01}$	$\mathbf{0.90}_{\pm 0.00}$	$0.91_{\pm 0.00}$	$0.90_{\pm 0.00}$	$0.86_{\pm 0.00}$	$\mathbf{0.93}_{\pm 0.00}$
	0.2	$0.65_{\pm 0.01}$	$0.55_{\pm 0.01}$	$0.61_{\pm 0.01}$	$\mathbf{0.74}_{\pm 0.01}$	$0.79_{\pm 0.01}$	$0.80_{\pm 0.01}$	$0.69_{\pm 0.01}$	$\mathbf{0.81}_{\pm 0.01}$	$0.83_{\pm 0.01}$	$0.79_{\pm 0.01}$	$0.76_{\pm 0.01}$	$\mathbf{0.86}_{\pm 0.01}$

→ **GDiff** posterior mean estimates **outperform baselines**

Conclusion

- Diffusion models to enable **posterior sampling** for **CMB component separation**
- **Cosmological inference** using a **Gibbs sampler** (GDiff)
  → allows to analyze CMB data with arbitrary diffusion-based foreground priors
- **GDiff**: Generic method for **blind denoising**

What's next? → Accelerating the inference
→ Beyond the Gaussian CMB assumption?

Thank you for your attention!

FLATIRON
INSTITUTE