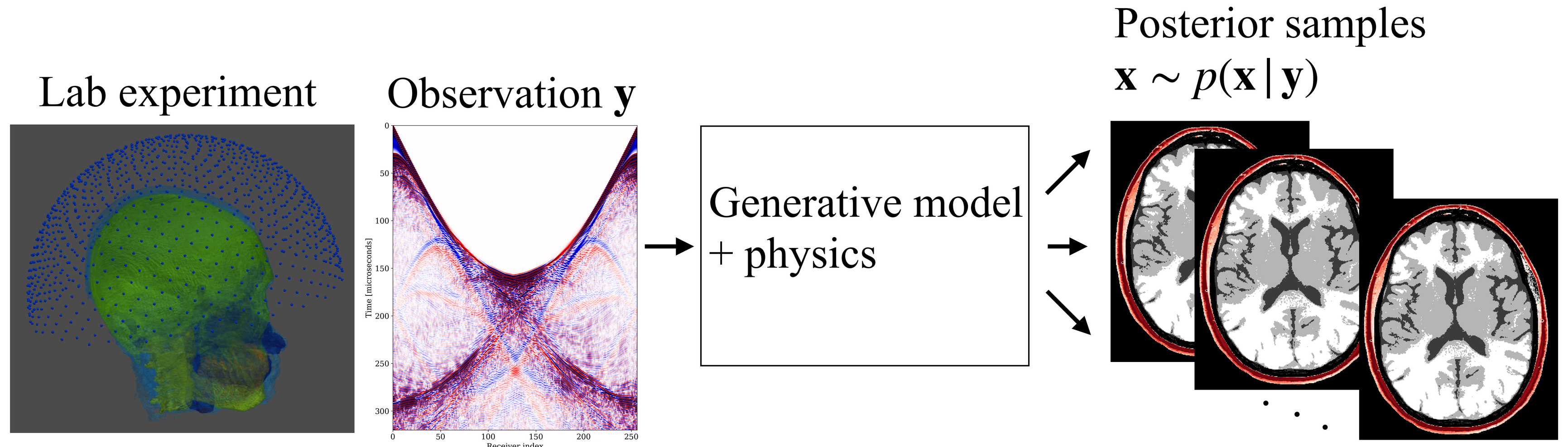
Refining Amortized Posterior Approximations using Gradient-Based Summary Statistics

Rafael Orozco[1], Ali Siahkoohi[2], Mathias Louboutin[1] and Felix J. Herrmann[1]
[1]Georgia Institute of Technology, [2]Rice University

Neural posterior approximation

Goal: Amortized posterior sampling method with low cost at inference but with better approximation quality than current amortized approaches.

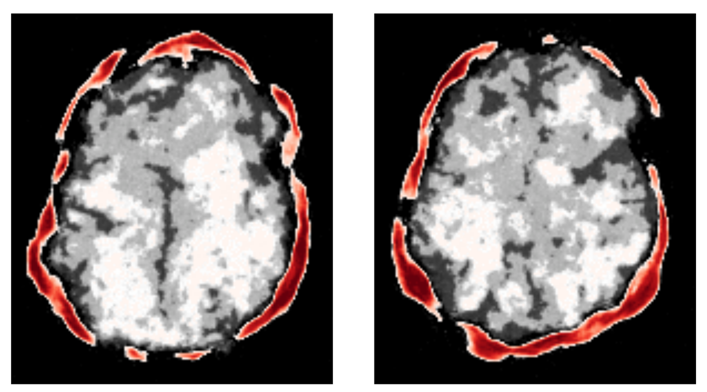


We start from conditional normalizing flows [1] that use a training objective requiring pairs of observations and ground truth image.

Problem: imaging modality is non-linear and high-dimensional thus posterior samples are poor when using raw observation $\mathbf{y}$ as input.

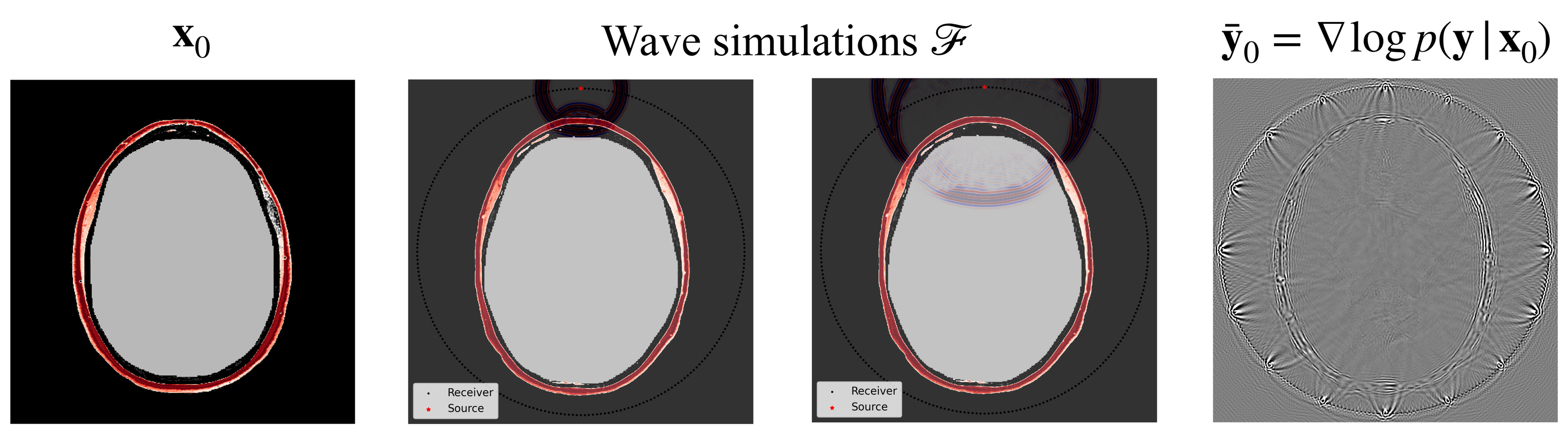
Posterior sampling after training:

$$\min_{\theta} \mathbb{E}_{\mathbf{x},\mathbf{y}\sim p(\mathbf{x},\mathbf{y})} \left[\left\| f_{\theta}(\mathbf{x};\mathbf{y}) \right\|_2^2 - \log|\det \mathrm{J}_{f_\theta}| \right]$$



Gradient-based summary statistic

Solution: use summary statistics to efficiently preprocess training of normalizing flow. The score of the log-likelihood $\log p(\mathbf{y}\,|\,\mathbf{x})$ requires solving wave simulations $\mathscr{F}$ but provides an informative summary statistic [2].



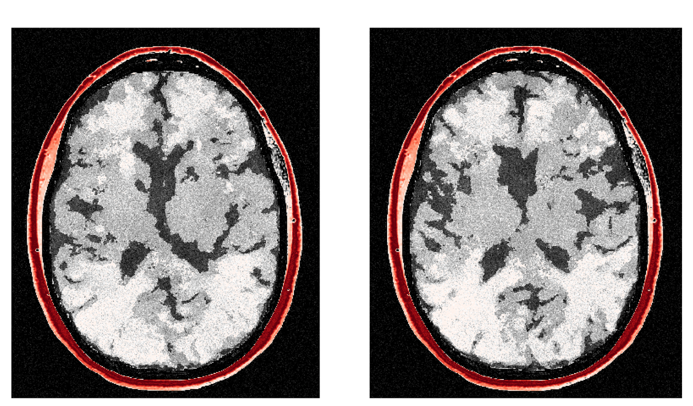
Note: this method still works in likelihood-free setting if you have a good enough approximation to the true likelihood. Here we assume Gaussian additive noise to form likelihood although the true likelihood corresponds to a Gaussian convolved with the transducer wavelet.

Problem: the quality of starting guess $\mathbf{x_0}$ might reduce the informativeness of the summary. We need a way to improve the starting guess.

Posterior sampling after training:

$$\min_{\theta_0} \mathbb{E}_{\mathbf{x},\mathbf{y}\sim p(\mathbf{x},\mathbf{y})} \left[\left\| f_{\theta_0}(\mathbf{x};\bar{\mathbf{y}}_0) \right\|_2^2 - \log|\det \mathrm{J}_{f_0}| \right] \quad (5)$$

where $\bar{\mathbf{y}}_0 = \nabla \log p(\mathbf{y}\,|\,\mathbf{x}_0) = \mathbf{J}_{\mathscr{F}}^{\top}(\mathscr{F}(\mathbf{x}_0) - \mathbf{y})$



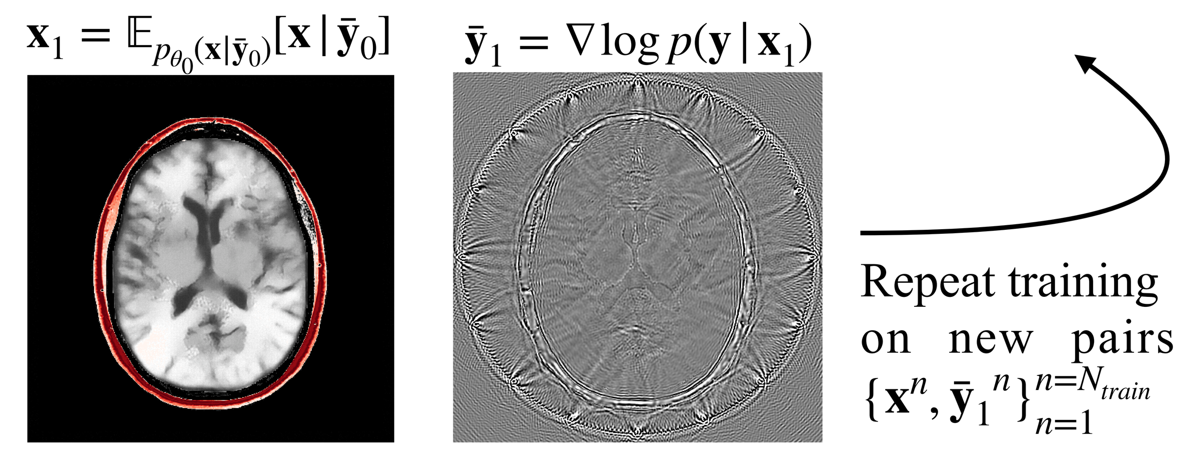
Proposed algorithm

Algorithm 2: Training Phase

```
Input: Paired observations {y^(n)}_{n=1}^{N_train} and corresponding parameters {x^(n)}_{n=1}^{N_train}
for n ← 1 to N_train do
    Generate starting guess: x_0^(n);
    Summarize observation with gradient: ȳ_0^(n) = ∇_x log p(y^(n) | x_0^(n));
    Add pairs to dataset: D_0^(n) = (x^(n), ȳ_0^(n));
end
for j = 0 to L do
    while conditional normalizing flow f_θj is not converged do
        Update θ_j to minimize objective function Equation (5) on dataset D_j;
    end
    for n = 1 to N_train do
        Update guess: x_{j+1}^(n) = E_{p_θ̂j(x|ȳ_j^(n))} [x | ȳ_j^(n)];
        Re-summarize observation with gradient: ȳ_{j+1}^(n) = ∇_x log p(y^(n) | x_{j+1}^(n));
        Add pairs to new dataset: D_{j+1}^(n) = (x^(n), ȳ_{j+1}^(n));
    end
end
Output: θ̂_0, θ̂_1, ..., θ̂_L Posterior Networks
```

Training phase for our medical imaging application:

- Train on pairs $\{\mathbf{x}^n, \bar{\mathbf{y}}_0{}^n\}_{n=1}^{n=N_{train}}$
- Produce posterior samples and calculate their mean to make next starting guess $\mathbf{x}_1$



Algorithm 1: Inference phase

```
Input: Observation y and starting guess x_0
for i = 0 to L − 1 do
    ȳ_i = ∇_x log p(y|x_i);
    x_{i+1} = E_{p_θ̂i(x|ȳ_i)} [x | ȳ_i];
end
ȳ_L = ∇_x log p(y|x_L);
Output: Final approximate posterior: p(x|y) ≈ p_θ̂L(x | ȳ_L)
```

TLDR: basically *probabilistic* loop-unrolling by making each unrolled layer be the posterior mean of a conditional normalizing flow.

Offline training cost: simulation cost × training samples × refinement steps L
Online inference cost: simulation cost × refinement steps L

As compared to a traditional least-squares, our posterior mean is more accurate at a fraction of the compute cost.

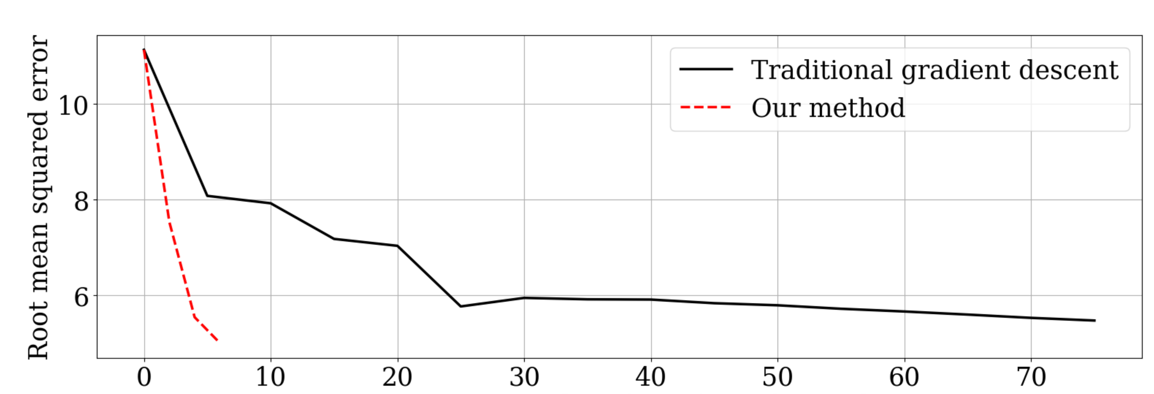
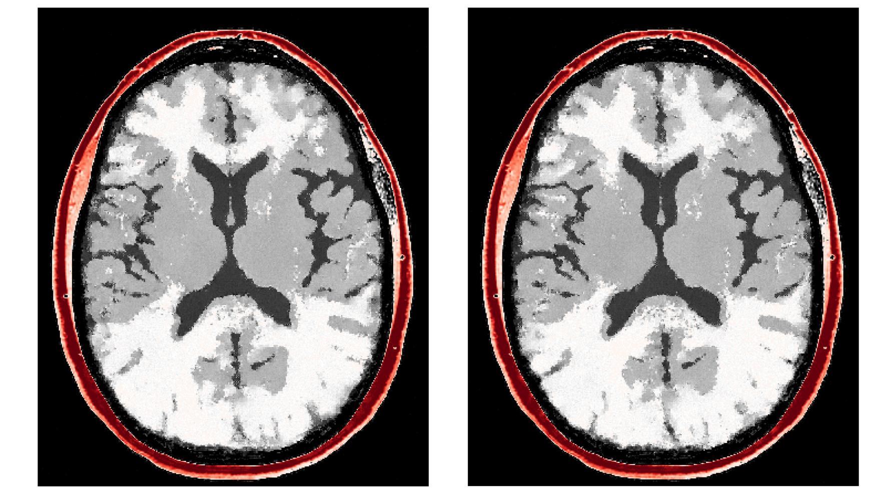


Image reconstruction

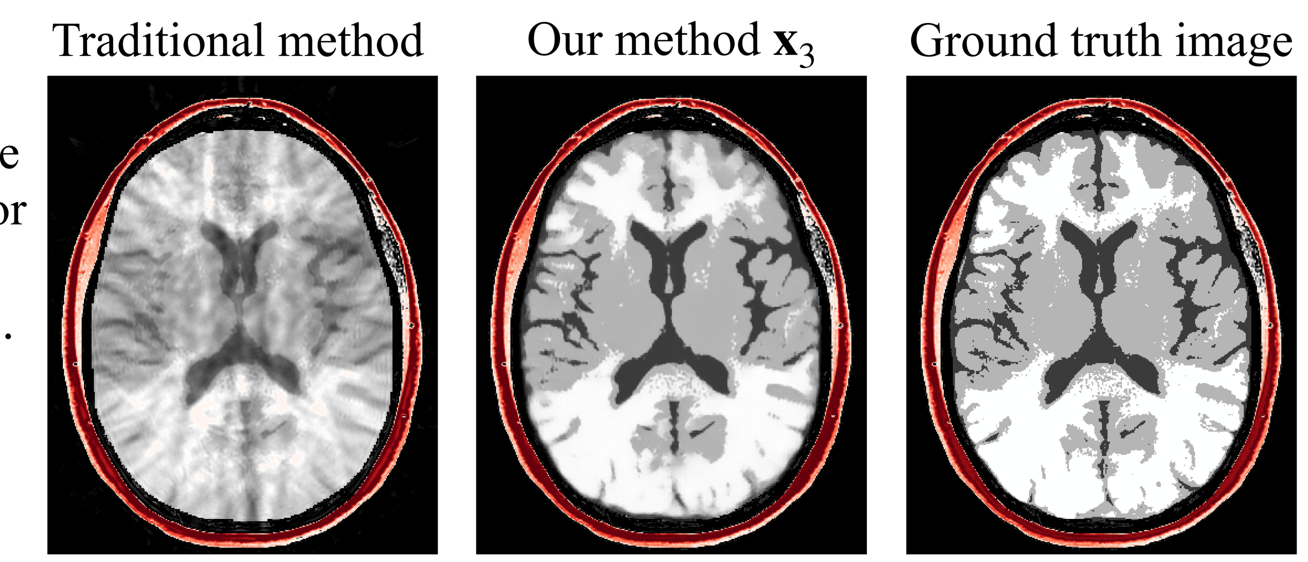
Posterior sampling after training:

$$\min_{\theta_2} \mathbb{E}_{\mathbf{x},\mathbf{y}\sim p(\mathbf{x},\mathbf{y})} \left[\left\| f_{\theta_2}(\mathbf{x};\bar{\mathbf{y}}_2) \right\|_2^2 - \log|\det \mathrm{J}_{f_\theta}| \right]$$

$$\bar{\mathbf{y}}_2 = \nabla \log p(\mathbf{y}\,|\,\mathbf{x}_2) = \mathbf{J}^{\top}(\mathscr{F}(\mathbf{x}_2) - \mathbf{y})$$

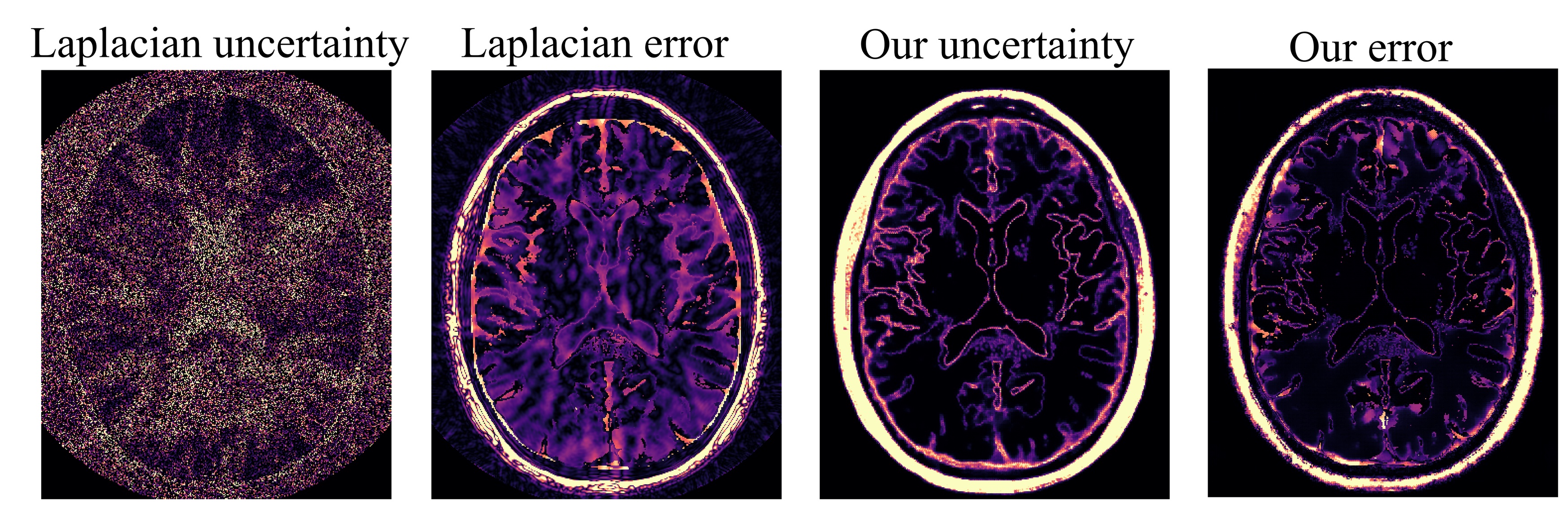


After three training iterations, we plot the mean of 256 posterior samples $\mathbf{x}_3 = \mathbb{E}_{p_{\theta_2}(\mathbf{x}|\bar{\mathbf{y}}_2)}[\mathbf{x}\,|\,\bar{\mathbf{y}}_2]$. The baseline is traditional least-squares.



Uncertainty

Compared to current UQ methods [3] (Laplacian approximation), our uncertainty is many times cheaper and yields interpretable UQ that correlates with errors.



Conclusions

By combining the statistical concept of summary statistics with machine learning, we arrive to an iterative framework that computes an efficient middle ground between amortized posterior approximation and non-amortized approaches.

Our method has an expensive offline training phase but is extremely efficient at online inference time as compared to traditional non-amortized approaches.

[1] Ardizzone, Lynton, et al. "Guided image generation with conditional invertible neural networks." (2019).
[2] Alsing, Justin, and Benjamin Wandelt. "Generalized massive optimal data compression." *Monthly Notices of the Royal Astronomical Society: Letters* 476.1 (2018):
[3] Bates, Oscar, et al. "A probabilistic approach to tomography,with an application to full waveform inversion in medical ultrasound." *Inverse Problems* (2022):