Stein Variational Descent Methods for Non-parametric Sampling in Inverse Problems

Felipe Suárez

Motivation

Bayesian Inverse Problems

$$\begin{cases} \nabla.(\theta\nabla u) = f \\ \dfrac{\partial u}{\partial n} = 0 \end{cases}$$

u partially known
f known
$\rightarrow \theta?$

$$\begin{cases} (\theta\omega^2 + \Delta)u = f \\ \dfrac{\partial u}{\partial n} = 0 \end{cases}$$

u partially known
f known
$\rightarrow \theta?$

Setup

Naïve Approach:

- Guess θ
- Solve the forward model

$$F(\theta) = u$$

- Iterate until $F(\theta)$ matches u measured

Bayesian Approach

$\theta \sim Prior$

$D = F(\theta) + \xi \longrightarrow$ Gaussian Noise

$\theta | D \sim Posterior \longrightarrow$

Likelihood Prior

$$\pi(\theta) = \frac{\mathbb{P}(D|\theta)\mathbb{P}(\theta)}{\mathbb{P}(D)}$$

Optimization Approach

$$\min_{\theta} L(F(\theta), D) + R(\theta)$$

- Infinite-dimensional Gradient Descent
- Adjoint-State method

Goal

$$\mathbb{E}_{\theta\sim\pi}[h(\theta)]$$

→ Low variance
→ Low sample complexity
→ Low cost per sample

Difficulties

- $\theta \in \mathbb{R}^n, n \gg 1$
- $\pi(\theta) \;\; \alpha \;\; \mathbb{P}(D|\theta)\mathbb{P}(\theta)$ → Un-normalized
- $\mathbb{P}(D|\theta)$ often costly

Purpose

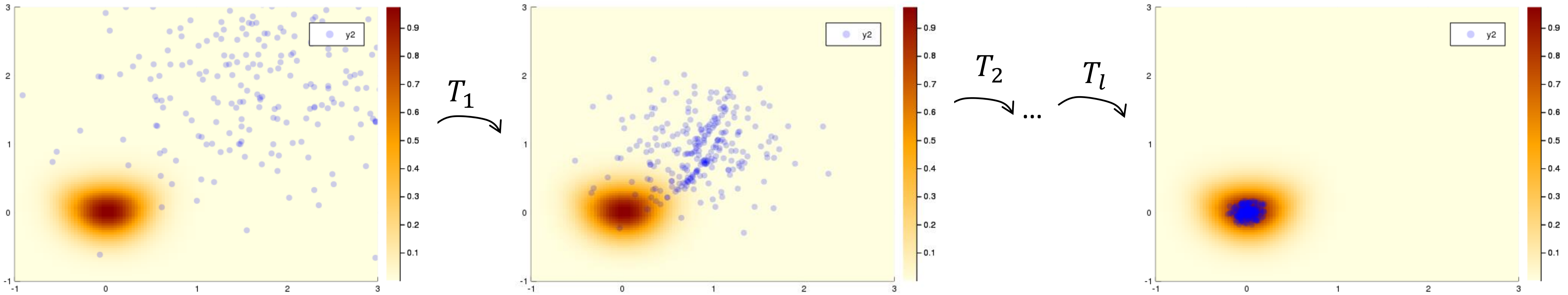
Design a tool for sampling π:

- *Particle transport* from a
- Cheaper distribution q toward π,
- Using cross-entropy minimization by
- A small *RKHS* perturbations and
- Simpler than MCMC! (Or not?)

Stein Method

$$g = \underset{h \in \langle k \rangle}{argmin} D_{KL}((I + h)_* q || \pi)$$

$$Z \xrightarrow{T} \pi$$



Three Key Lemmas

$$\frac{\delta}{\delta h} D_{KL}((I+h)_* q \,||\, p) \Big|_{h=0} = -tr\mathbb{E}_q \mathbb{D}_p h$$

Steepest Descent

$$\underset{h \in \langle k \rangle}{argmax} \; tr\mathbb{E}_q \mathbb{D}_p h = \mathbb{E}_q \mathbb{D}_p k(x,\cdot)$$

$$\frac{\delta^2}{\delta h \delta w} D_{KL}((I+h+w)_* q \,||\, p) = -tr\mathbb{E}_q[\nabla^2 \log p \; h \, g^T \; - \nabla h \, \nabla g]$$

Newton Iteration

$$HJ(h,g) = \nabla \mathrm{J}(\mathrm{h}) \qquad h \in RKHS$$

Algorithm

Input: Sample $\{x_i\}_i \sim q$, target π, kernel k
Output: Mapped Sample $\{Tx_i\}_i \sim \pi$

For $i = 1, \dots, n$

1. Compute a Descent direction $G(x_i)$
2. Apply transport map

$$T = I - \eta G$$

endFor

First Order

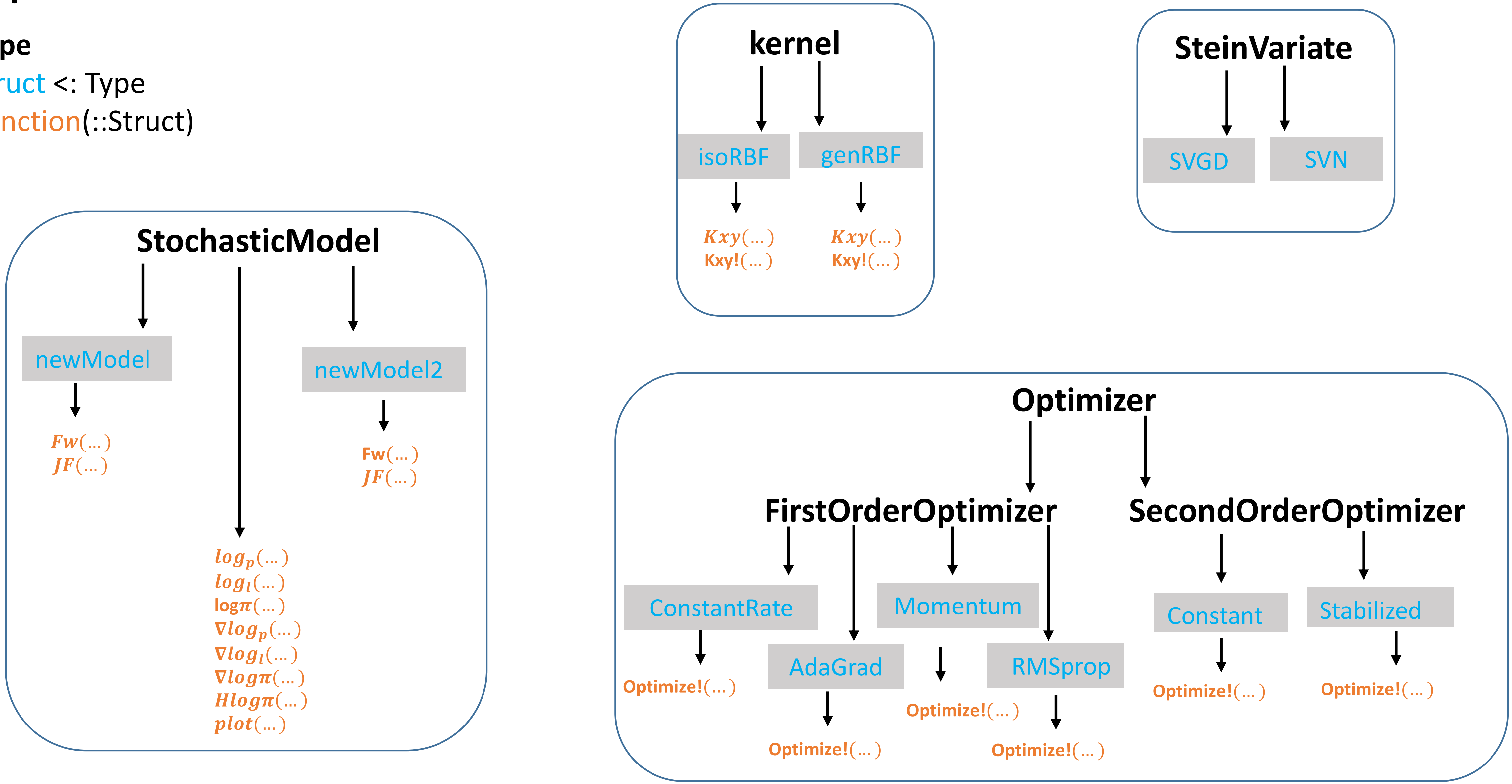
- Steepest Descent
- GD with Momentum
- AdaGrad
- RMSprop

Second Order(ish)

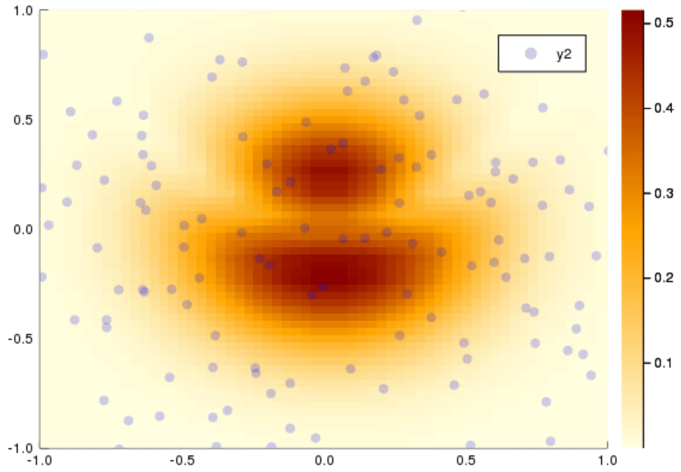
- Gauss Newton

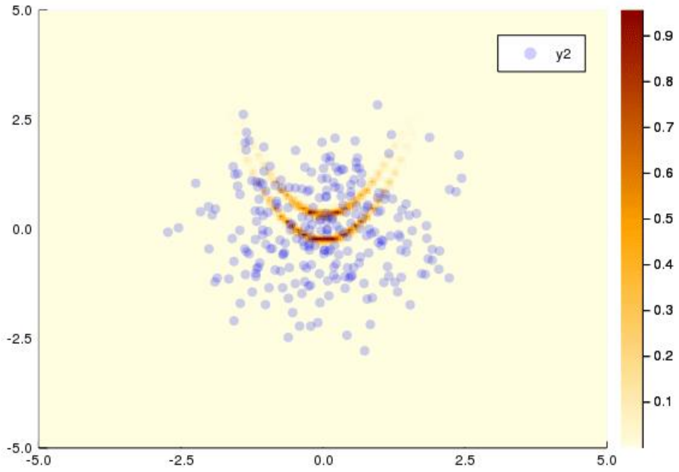
Implementation in Julia

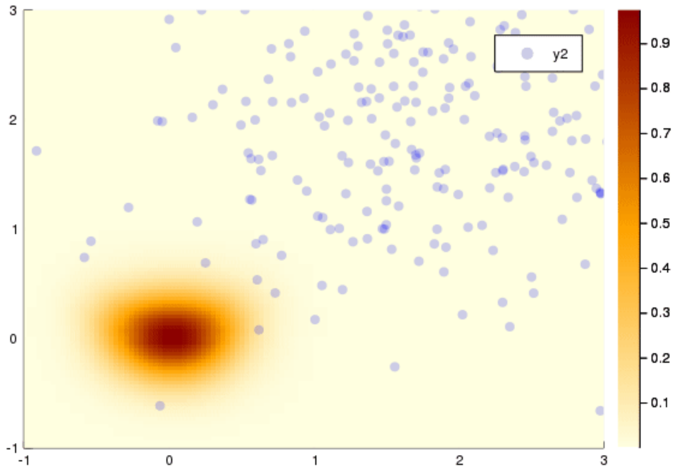
Type
Struct <: Type
Function(::Struct)



Some Results







References
[1] Liu, Qiang, and Dilin Wang. "Stein variational gradient descent: A general purpose bayesian inference algorithm." *Advances In Neural Information Processing Systems. 2016.*
[2] Detommaso, G., Cui, T., **Marzouk**, Y., Spantini, A., & Scheichl, R. (2018). A Stein variational Newton method. In *Advances in Neural Information Processing Systems* (pp. 9186-9196).
[3] Chen, Wilson Ye, et al. "Stein points." *arXiv preprint arXiv:1803.10161 (2018).*