Monte Carlo with the repelled Poisson point process

Diala Hawat, Raphaël Lachièze-Rey, and Rémi Bardenet

Université de Lille, CNRS, Centrale Lille ; UMR 9189 – CRIStAL, F-59000 Lille, France.
Université Paris Cité, Map5, Paris, France.

Contents

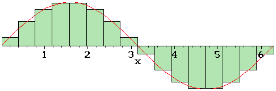
Numerical integration

Let f be a continuous fonction supported on a compact K.

Numerical integration

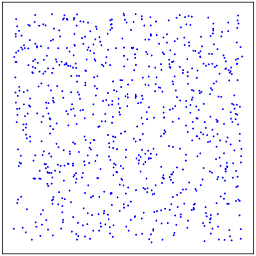
Let f be a continuous fonction supported on a compact K.

- Estimating: $\int_K f(\mathbf{x})d\mathbf{x} \approx \sum_{i=1}^{N} w_i f(\mathbf{x}_i)$



Numerical integration

Let $\mathcal{X}$ be a (simple) point process of intensity ρ



Numerical integration

Let $\mathcal{X}$ be a (simple) point process of intensity ρ

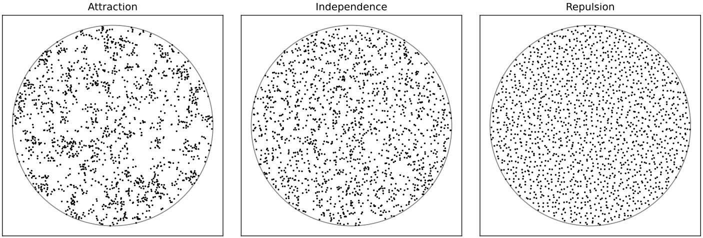
$$\mathbb{E}\left[\sum_{\mathbf{x}\in\mathcal{X}\cap K}\frac{1}{\rho(\mathbf{x})}f(\mathbf{x})\right]=\int_K f(\mathbf{x})d\mathbf{x}.$$

Numerical integration

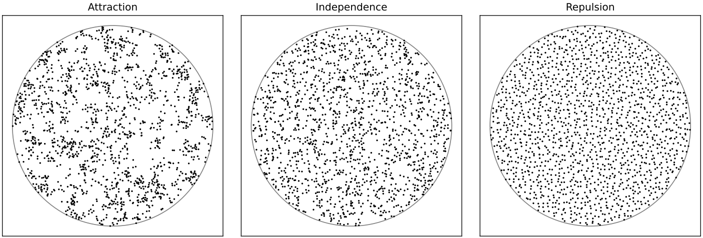
Let $\mathcal{X}$ be a (simple) point process of intensity ρ

$$\mathbb{E}\Big[\underbrace{\sum_{\mathbf{x} \in \mathcal{X} \cap K} \frac{1}{\rho(\mathbf{x})} f(\mathbf{x})}_{\text{Unbiased estimator } := \widehat{I}(f)} \Big] = \int_K f(\mathbf{x}) d\mathbf{x}.$$

$$\widehat{I}(f) = \sum_{\mathbf{x}\in\mathcal{X}\cap K} \frac{1}{\rho(\mathbf{x})} f(\mathbf{x}) \approx \int_K f(\mathbf{x}) d\mathbf{x}$$



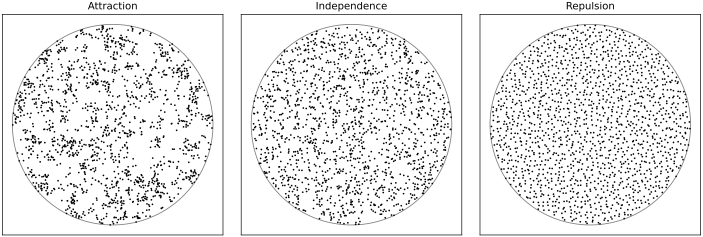
$$\widehat{I}(f) = \sum_{\mathbf{x}\in\mathcal{X}\cap K} \frac{1}{\rho(\mathbf{x})} f(\mathbf{x}) \approx \int_K f(\mathbf{x}) d\mathbf{x}$$



Simple Monte Carlo:

- $\mathcal{X}$ a homogeneous Poisson point porcess (PPP) of intensity ρ of $\mathbb{R}^d$.
- Sampling from $\mathcal{X}$ is fast.
- $\mathbb{V}\mathrm{ar}[\widehat{I}(f)] = \frac{\int_K f^2(\mathbf{x}) d\mathbf{x}}{\rho} = \mathcal{O}(N^{-1})$.

A. B. Owen. Monte Carlo theory, methods and examples. 2013.



- Monte Carlo with DPP [1] [2] [3] : Convergence rate $\mathcal{O}(N^{-1-1/d})$ (with PPP we had $\mathcal{O}(N^{-1})$).
- Sampling from the DPP is expensive ($\sim \mathcal{O}(N^3)$).

[1] S. M. Ermakov and V. G. Zolotukhin. Polynomial approximations and the monte- carlo method. Theory of Probability & Its Applications, 1960.

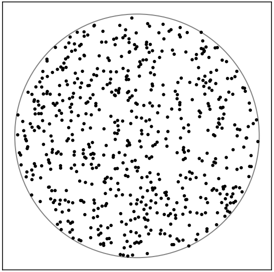
[2] R. Bardenet and A. Hardy. Monte Carlo with determinantal point processes. The Annals of Applied Probability, 2020.

[3] J.-F. Coeurjolly, A. Mazoyer, and P.-O. Amblard. Monte Carlo integration of non-differentiable functions on $[0, 1]^i$, $i = 1, \ldots, d$, using a single determinantal point pattern defined on $[0, 1]^d$. Electronic Journal of Statistics, 2021.

Repelled point process

Construction

$\mathcal{X}$ a (simple) stationary point process of intensity ρ of $\mathbb{R}^d$.



Poisson

Construction

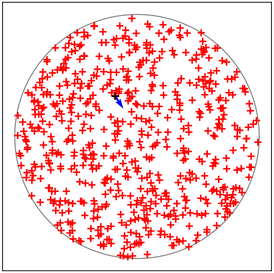
$\mathcal{X}$ a (simple) stationary point process of intensity ρ of $\mathbb{R}^d$.

- Force :

$$F_{\mathcal{X}}(\mathbf{x}) := \sum_{\substack{\mathbf{z}\in\mathcal{X}\setminus\{\mathbf{x}\} \\ \|\mathbf{x}-\mathbf{z}\|_2\uparrow}} \frac{\mathbf{x}-\mathbf{z}}{\|\mathbf{x}-\mathbf{z}\|_2^d}.$$

- Repulsion operator:

$$\Pi_\varepsilon : \mathcal{X} \longmapsto \{\mathbf{x} + \varepsilon F_{\mathcal{X}}(\mathbf{x}) : \mathbf{x} \in \mathcal{X}\}.$$



Poisson

Chatterjee, Sourav, Ron Peled, Yuval Peres, and Dan Romik. "Gravitational Allocation to Poisson Points." Annals of Mathematics, 2010.

Construction

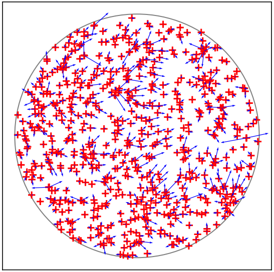
$\mathcal{X}$ a (simple) stationary point process of intensity ρ of $\mathbb{R}^d$.

- Force :

$$F_{\mathcal{X}}(\mathbf{x}) := \sum_{\substack{\mathbf{z}\in\mathcal{X}\setminus\{\mathbf{x}\} \\ \|\mathbf{x}-\mathbf{z}\|_2\uparrow}} \frac{\mathbf{x}-\mathbf{z}}{\|\mathbf{x}-\mathbf{z}\|_2^d}.$$

- Repulsion operator:

$$\Pi_\varepsilon : \mathcal{X} \longmapsto \{\mathbf{x} + \varepsilon F_{\mathcal{X}}(\mathbf{x}) : \mathbf{x} \in \mathcal{X}\}.$$



Poisson

Construction

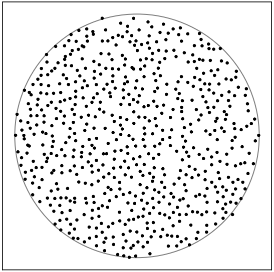
$\mathcal{X}$ a (simple) stationary point process of intensity ρ of $\mathbb{R}^d$.

- Force :

$$F_{\mathcal{X}}(\mathbf{x}) := \sum_{\substack{\mathbf{z}\in\mathcal{X}\setminus\{\mathbf{x}\} \\ \|\mathbf{x}-\mathbf{z}\|_2\uparrow}} \frac{\mathbf{x}-\mathbf{z}}{\|\mathbf{x}-\mathbf{z}\|_2^d}.$$

- Repulsion operator:

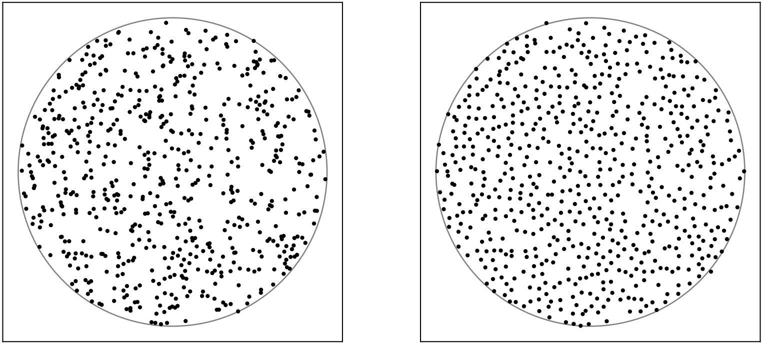
$$\Pi_\varepsilon : \mathcal{X} \longmapsto \{\mathbf{x} + \varepsilon F_{\mathcal{X}}(\mathbf{x}) : \mathbf{x} \in \mathcal{X}\}.$$



Repelled Poisson

Construction

$\mathcal{X}$ a (simple) stationary point process of intensity ρ of $\mathbb{R}^d$.



Poisson | Repelled Poisson

Main result

Let $\mathcal{P} \in \mathbb{R}^d$ be a PPP of intensity $\rho > 0$ with $d \geq 3$.

- For any $\varepsilon \in \mathbb{R}$, and any two distinct points $\mathbf{x}$, $\mathbf{y} \in \mathcal{P}$, we have a.s.

$$\mathbf{x} + \varepsilon F_{\mathcal{P}}(\mathbf{x}) \neq \mathbf{y} + \varepsilon F_{\mathcal{P}}(\mathbf{y}).$$

  Moreover, $\Pi_\varepsilon \mathcal{P}$ is a stationary and isotropic point process of intensity ρ.

Main result

Let $\mathcal{P} \in \mathbb{R}^d$ be a PPP of intensity $\rho > 0$ with $d \geq 3$.

- Let $\varepsilon \in (-1, 1)$ and $R > 0$. For any positive integer m,

$$\mathbb{E}\left[\left(\sum_{\mathbf{x} \in \Pi_\varepsilon \mathcal{P}} \mathbb{1}_{B(0,R)}(\mathbf{x})\right)^m\right] < \infty.$$

Main result

Let $\mathcal{P} \in \mathbb{R}^d$ be a PPP of intensity $\rho > 0$ with $d \geq 3$.

- Let $\varepsilon \in (-1, 1)$. For any function $f \in C^2(\mathbb{R}^d)$ of compact support K, we have

$$\mathbb{V}ar\left[\widehat{I}_{\Pi_\varepsilon \mathcal{P}}(f)\right] = \mathbb{V}ar\left[\widehat{I}_{\mathcal{P}}(f)\right](1 - 2d\kappa_d \rho\varepsilon) + \mathcal{O}(\varepsilon^2).$$

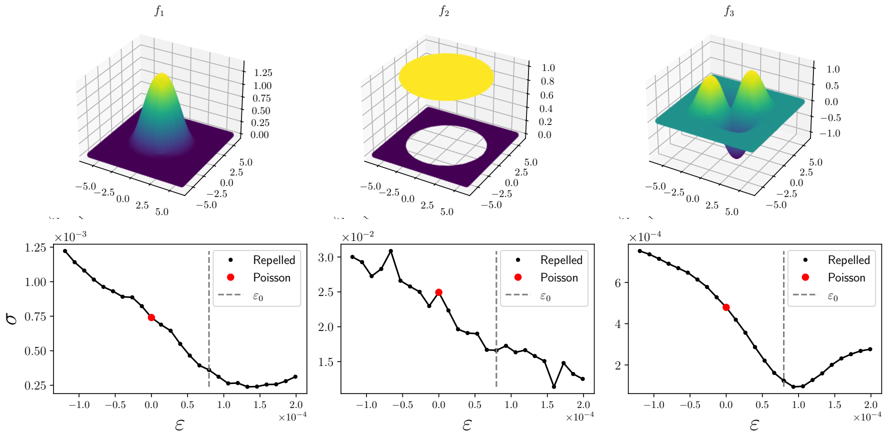
- For $\varepsilon > 0$ small enough,

$$\mathbb{V}ar\left[\widehat{I}_{\Pi_\varepsilon \mathcal{P}}(f)\right] < \mathbb{V}ar\left[\widehat{I}_{\mathcal{P}}(f)\right].$$

- Computational complexity $\mathcal{O}(N^2)$ (parallalizable).

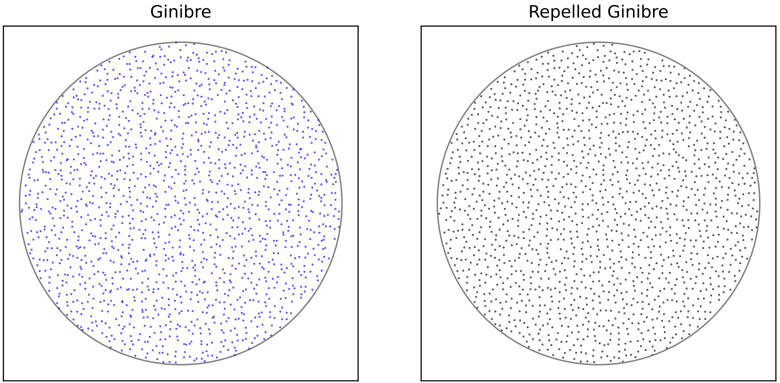
Experiment

Experiment



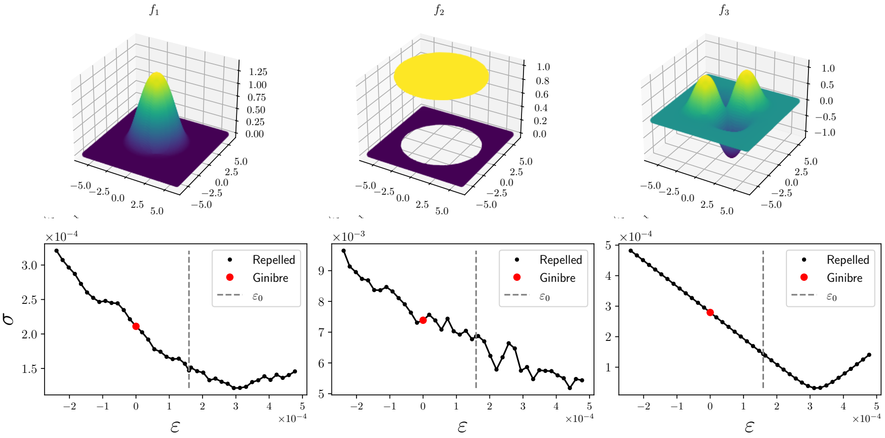
d=3

Other models



D. Hawat, G. Gautier, R. Bardenet, R. Lachièze-Rey. *On estimating the structure factor of a point process, with applications to hyperuniformity*. Statistics and Computing, 2023.

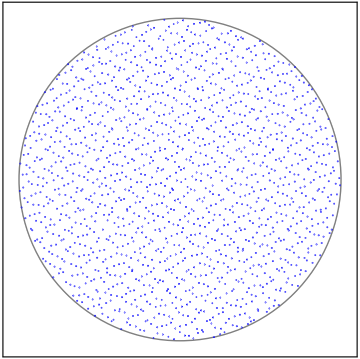
Other models



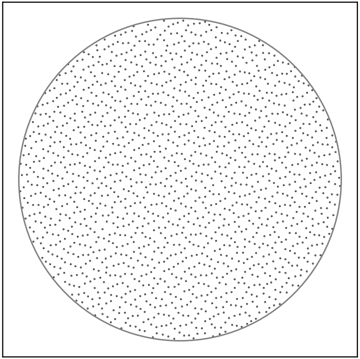
Variance $d = 2$

Other models

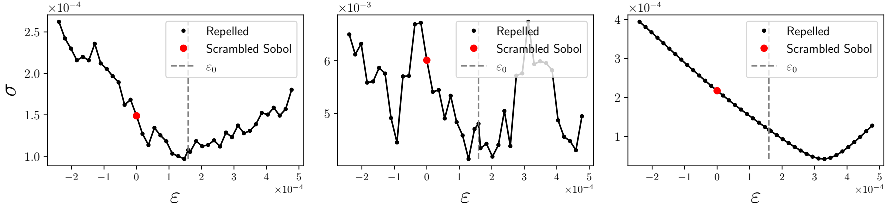
Scrambled Sobol



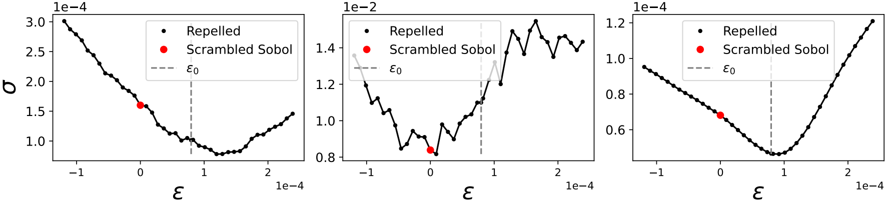
Repelled Scrambled Sobol



Other models



Variance $d = 2$



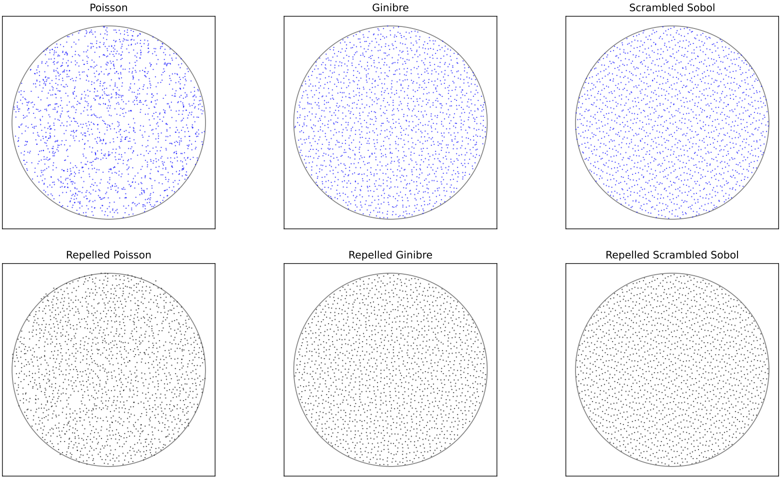
Variance $d = 3$

Conclusion

Conclusion

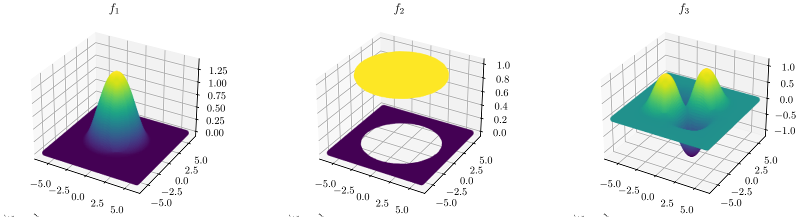
- Monte Carlo variance reduction with the repelled Poisson point process
- Open source Python toolbox `MCRPPy`

THANK YOU

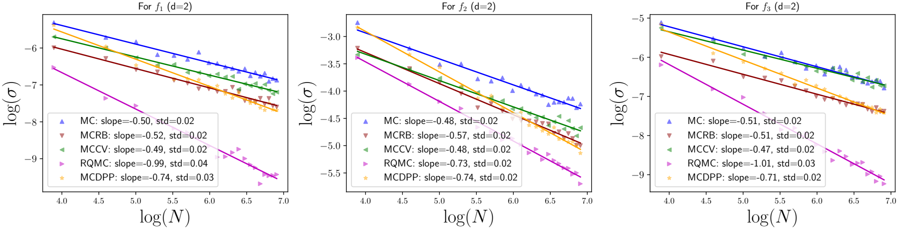


Webpage: `https://dhawat.github.io/`

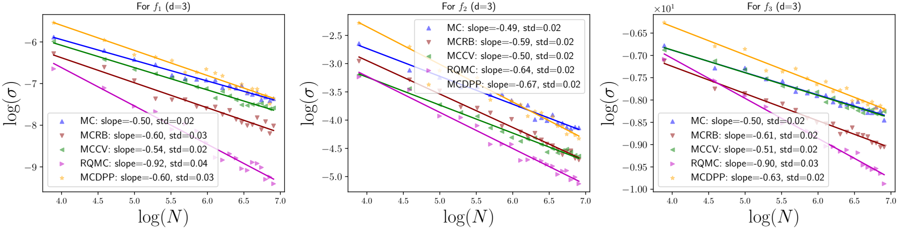
Experiment


f_1
1.25
1.00
0.75
0.50
0.25
0.00
−5.0
−2.5
0.0
2.5
5.0
f_2
1.0
0.8
0.6
0.4
0.2
0.0
f_3
1.0
0.5
0.0
−0.5
−1.0

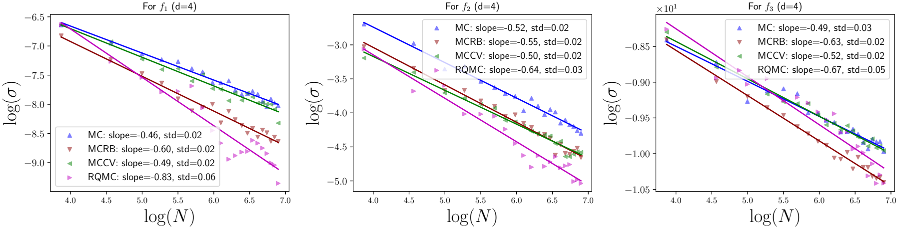
Experiment


For f_1 (d=2)
log(σ)
log(N)
MC: slope=-0.50, std=0.02
MCRB: slope=-0.52, std=0.02
MCCV: slope=-0.49, std=0.02
RQMC: slope=-0.99, std=0.04
MCDPP: slope=-0.74, std=0.03
For f_2 (d=2)
log(σ)
log(N)
MC: slope=-0.48, std=0.02
MCRB: slope=-0.57, std=0.02
MCCV: slope=-0.48, std=0.02
RQMC: slope=-0.73, std=0.02
MCDPP: slope=-0.74, std=0.02
For f_3 (d=2)
log(σ)
log(N)
MC: slope=-0.51, std=0.02
MCRB: slope=-0.51, std=0.02
MCCV: slope=-0.47, std=0.02
RQMC: slope=-1.01, std=0.03
MCDPP: slope=-0.71, std=0.02

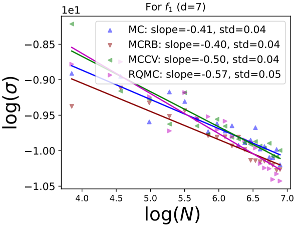
Experiment

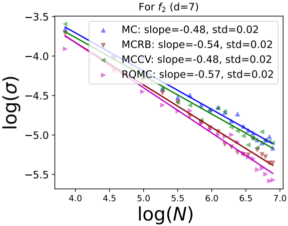

For f_1 (d=3)
MC: slope=-0.50, std=0.02
MCRB: slope=-0.60, std=0.03
MCCV: slope=-0.54, std=0.02
RQMC: slope=-0.92, std=0.04
MCDPP: slope=-0.60, std=0.03
For f_2 (d=3)
MC: slope=-0.49, std=0.02
MCRB: slope=-0.59, std=0.02
MCCV: slope=-0.50, std=0.02
RQMC: slope=-0.64, std=0.02
MCDPP: slope=-0.67, std=0.02
For f_3 (d=3)
$\times 10^1$
MC: slope=-0.50, std=0.02
MCRB: slope=-0.61, std=0.02
MCCV: slope=-0.51, std=0.02
RQMC: slope=-0.90, std=0.03
MCDPP: slope=-0.63, std=0.02
$\log(\sigma)$
$\log(N)$

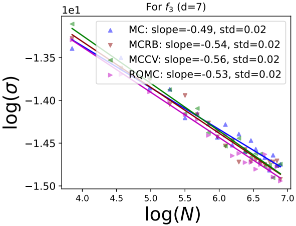
Experiment


For f_1 (d=4)
MC: slope=-0.46, std=0.02
MCRB: slope=-0.60, std=0.02
MCCV: slope=-0.49, std=0.02
RQMC: slope=-0.83, std=0.06
$\log(\sigma)$
$\log(N)$
For f_2 (d=4)
MC: slope=-0.52, std=0.02
MCRB: slope=-0.55, std=0.02
MCCV: slope=-0.50, std=0.02
RQMC: slope=-0.64, std=0.03
For f_3 (d=4)
$\times 10^1$
MC: slope=-0.49, std=0.03
MCRB: slope=-0.63, std=0.02
MCCV: slope=-0.52, std=0.02
RQMC: slope=-0.67, std=0.05

Experiment


For f_1 (d=7)
1e1
MC: slope=-0.41, std=0.04
MCRB: slope=-0.40, std=0.04
MCCV: slope=-0.50, std=0.04
RQMC: slope=-0.57, std=0.05
log(σ)
log(N)


For f_2 (d=7)
MC: slope=-0.48, std=0.02
MCRB: slope=-0.54, std=0.02
MCCV: slope=-0.48, std=0.02
RQMC: slope=-0.57, std=0.02
log(σ)
log(N)


For f_3 (d=7)
1e1
MC: slope=-0.49, std=0.02
MCRB: slope=-0.54, std=0.02
MCCV: slope=-0.56, std=0.02
RQMC: slope=-0.53, std=0.02
log(σ)
log(N)