Elliptic PDE learning is provably data-efficient

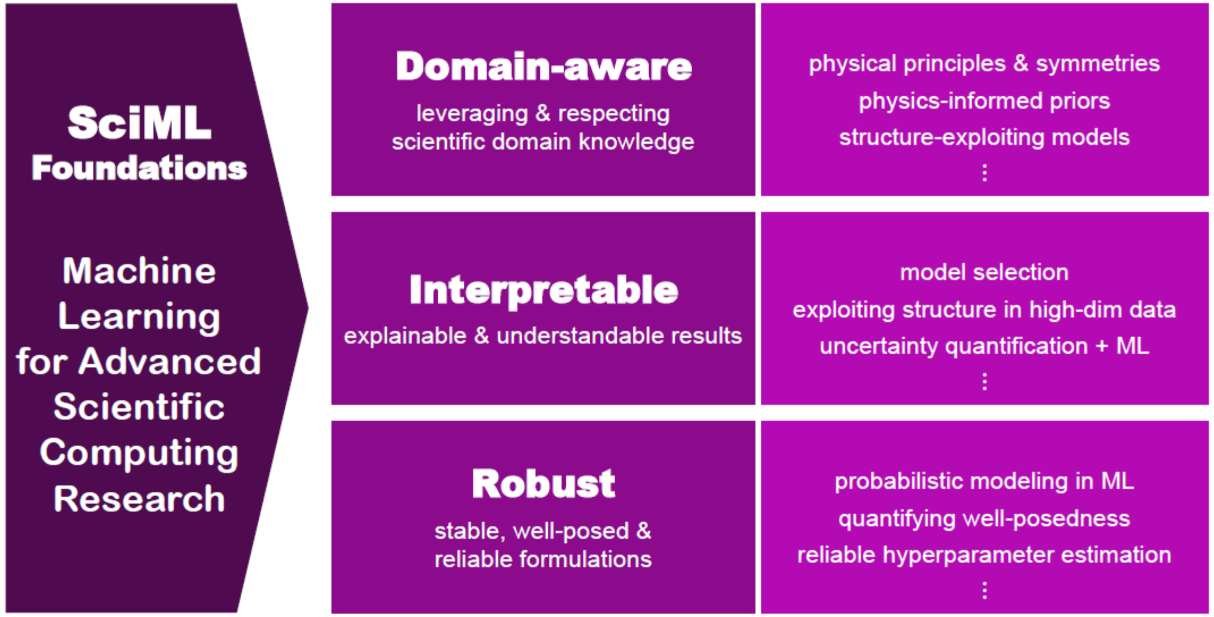
Nicolas Boullé

Diana Halikias

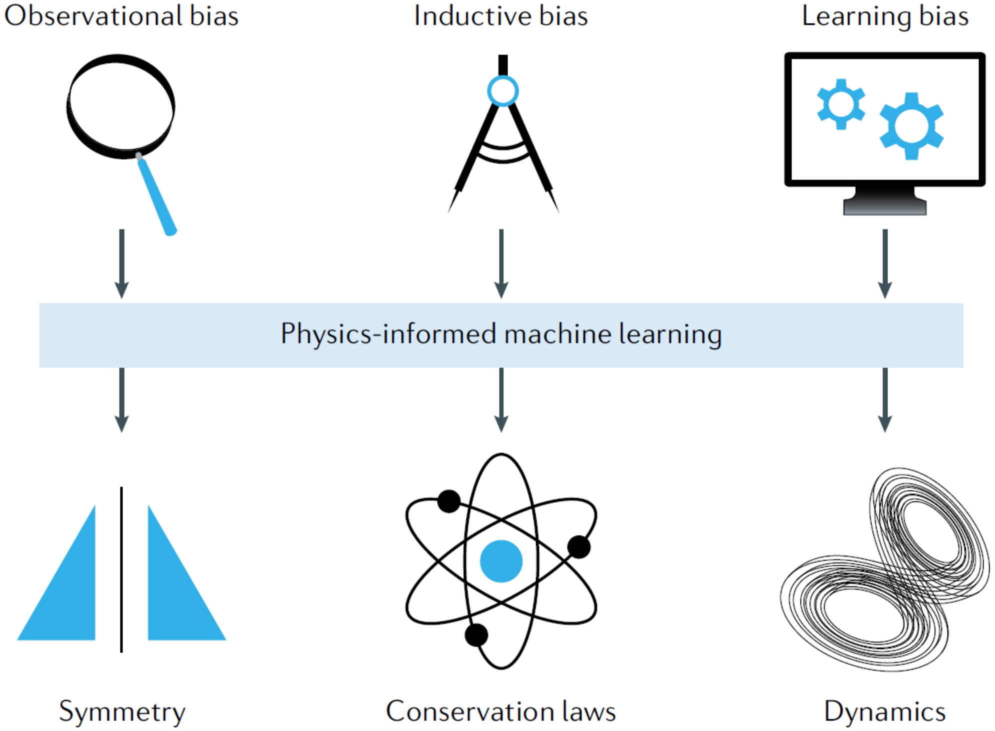
Alex Townsend

UNIVERSITY OF CAMBRIDGE

Scientific machine learning



[Baker et al., DOE Tech. Report, 2019]

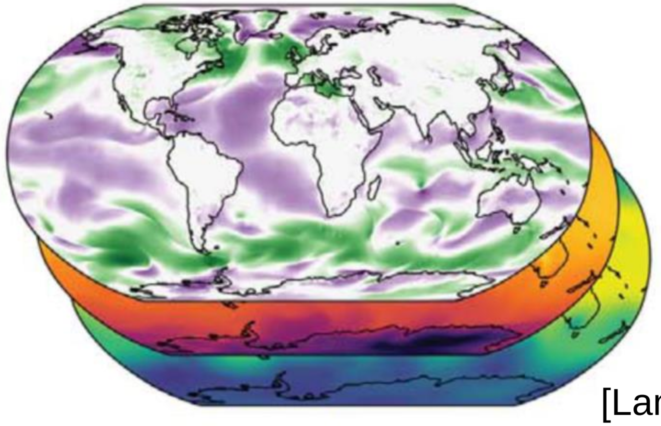


[Karniadakis et al., 2021]

Combining numerical analysis and machine learning for scientific discoveries.

Recent applications of SciML

Weather forecasting

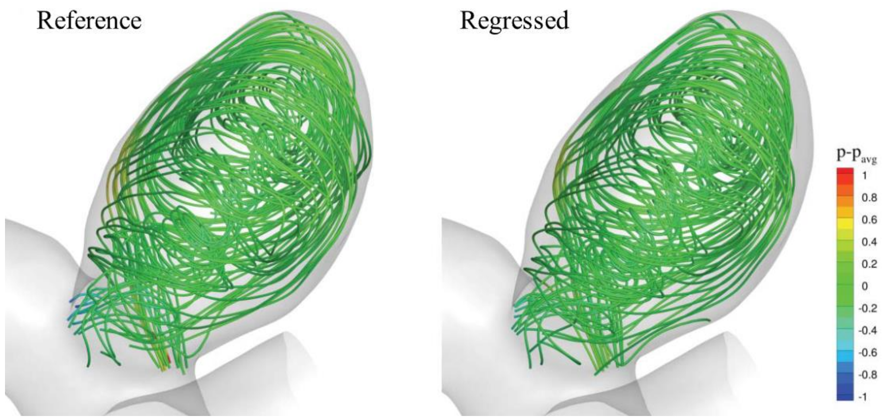


[Lam et al., Science, 2023]

Protein folding

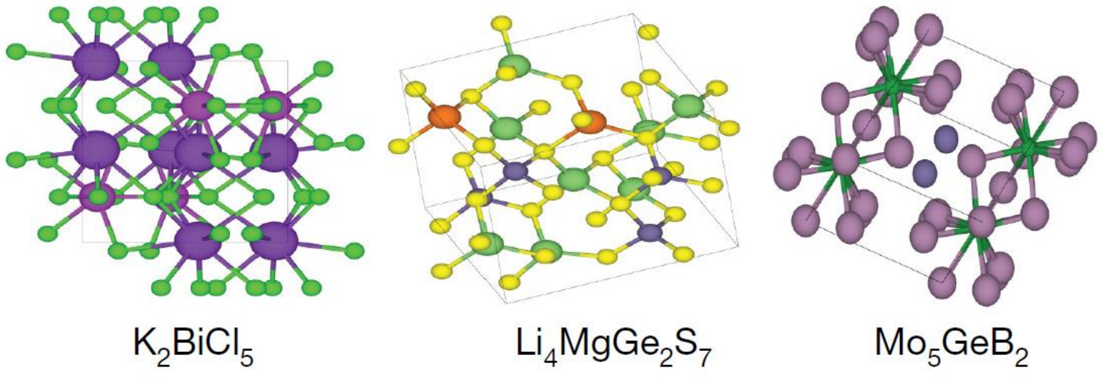
[Jumper et al., Nature, 2021]

Numerical simulations



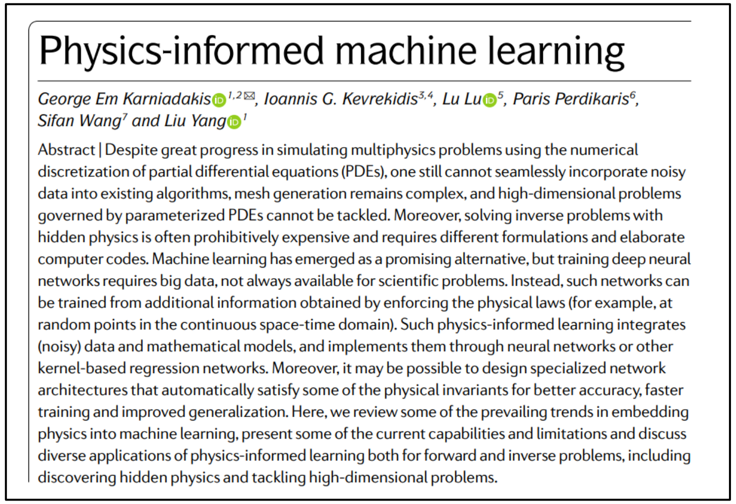
[Raissi, Yazdani, Karniadakis, Science, 2020]

Materials discovery



[Merchant et al., Nature, 2023]

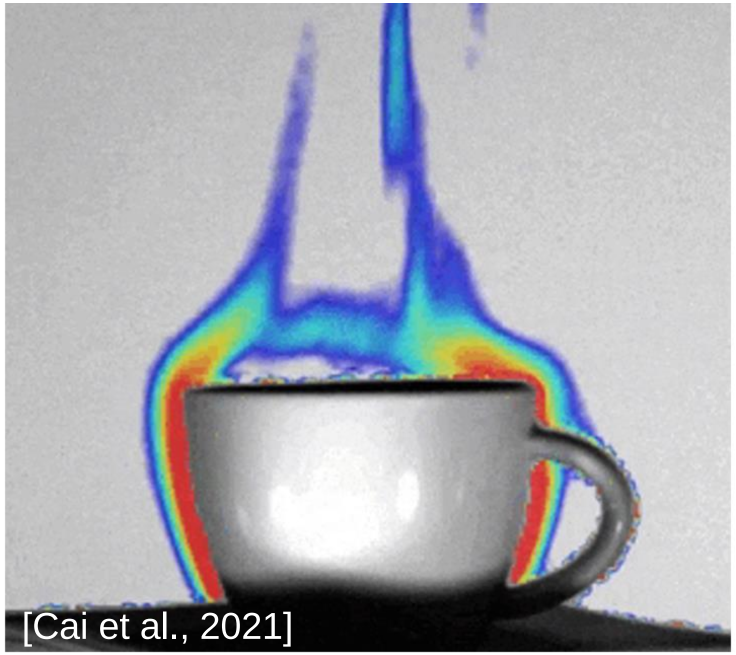
Operator learning



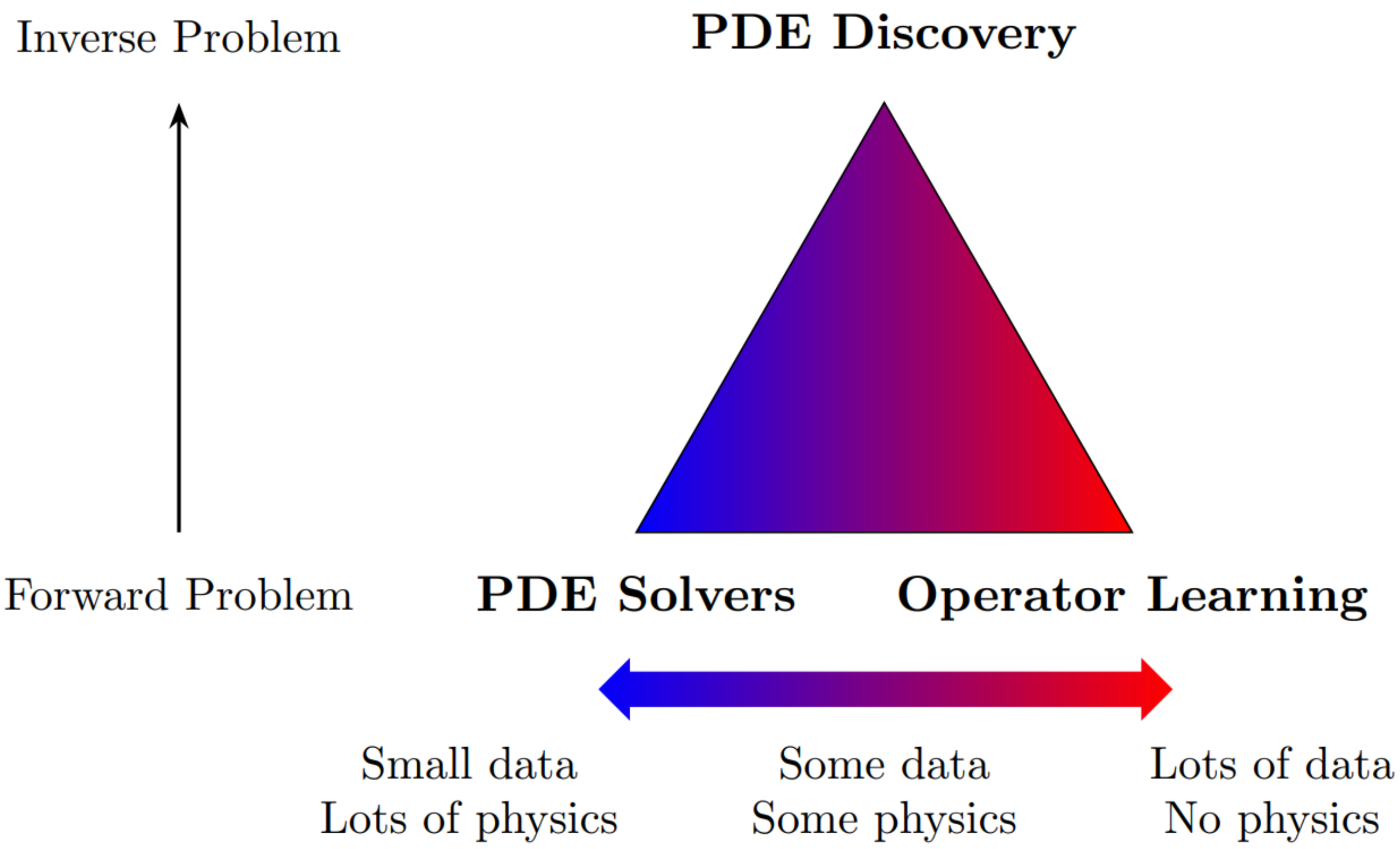
Physics-informed machine learning

George Em Karniadakis[1,2], Ioannis G. Kevrekidis[3,4], Lu Lu[5], Paris Perdikaris[6], Sifan Wang[7] and Liu Yang[1]

Abstract | Despite great progress in simulating multiphysics problems using the numerical discretization of partial differential equations (PDEs), one still cannot seamlessly incorporate noisy data into existing algorithms, mesh generation remains complex, and high-dimensional problems governed by parameterized PDEs cannot be tackled. Moreover, solving inverse problems with hidden physics is often prohibitively expensive and requires different formulations and elaborate computer codes. Machine learning has emerged as a promising alternative, but training deep neural networks requires big data, not always available for scientific problems. Instead, such networks can be trained from additional information obtained by enforcing the physical laws (for example, at random points in the continuous space-time domain). Such physics-informed learning integrates (noisy) data and mathematical models, and implements them through neural networks or other kernel-based regression networks. Moreover, it may be possible to design specialized network architectures that automatically satisfy some of the physical invariants for better accuracy, faster training and improved generalization. Here, we review some of the prevailing trends in embedding physics into machine learning, present some of the current capabilities and limitations and discuss diverse applications of physics-informed learning both for forward and inverse problems, including discovering hidden physics and tackling high-dimensional problems.



[Cai et al., 2021]

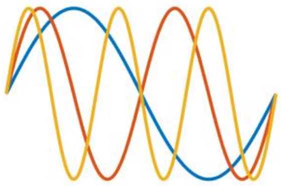


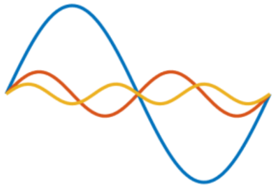
Recent survey: B., Townsend, ''A Mathematical Guide to Operator Learning'', 2023.

Introduction

Aim: Learn the solution operator of unknown linear PDEs $\mathcal{L}(u) = f$ from observation data:

f → Learn $\mathcal{L}^{-1}$ → u

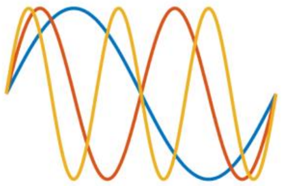


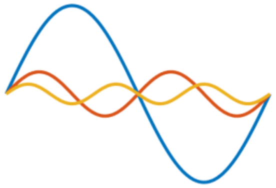


Introduction

Aim: Learn the solution operator of unknown linear PDEs $\mathcal{L}(u) = f$ from observation data:

f → Learn $\mathcal{L}^{-1}$ → u





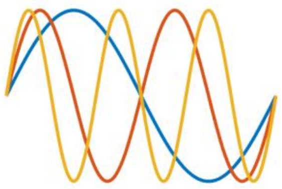
Main contributions:

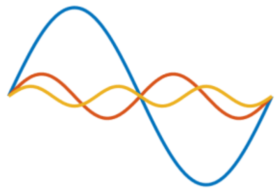
- Theoretical result quantifying the number of training pairs (f, u) needed
- A practical deep learning approach to learn Green's functions

Introduction

Aim: Learn the solution operator of unknown linear PDEs $\mathcal{L}(u) = f$ from observation data:

f → Learn $\mathcal{L}^{-1}$ → u



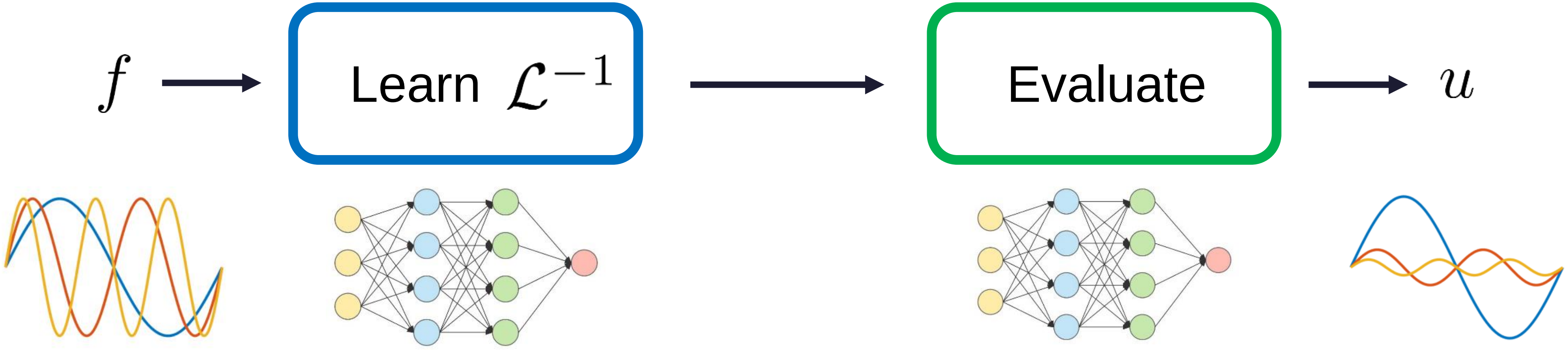


Key ideas:

- Randomized numerical linear algebra
- Regularity of Green's functions
- Rational neural networks

Standard approaches of PDE learning

Approximating the solution operator $\mathcal{L}^{-1}(f) = u$

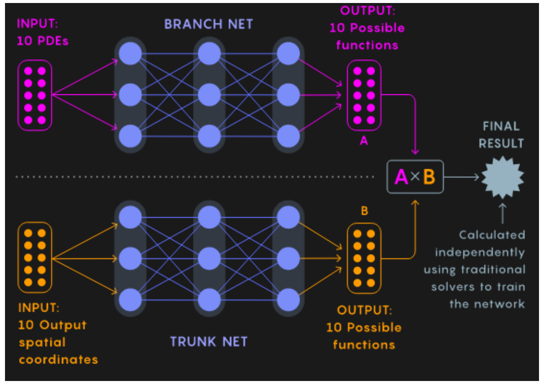


Standard approaches of PDE learning

Approximating the solution operator $\mathcal{L}^{-1}(f) = u$

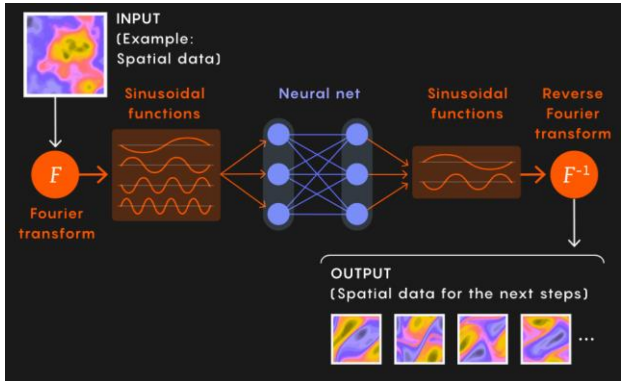
f → **Learn $\mathcal{L}^{-1}$** → **Evaluate** → u

DeepONet



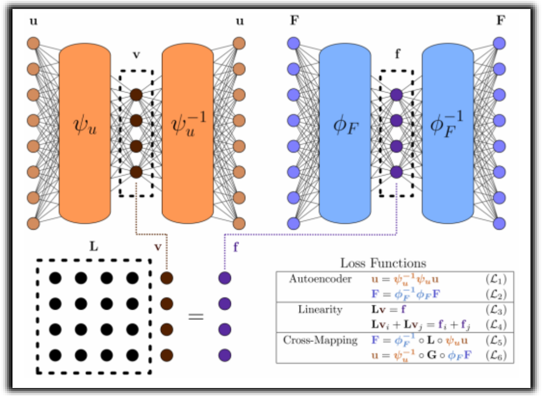
[Quanta Magazine; Lu et al, 2021]

Fourier Neural Operator



[Quanta Magazine; Li et al, 2020]

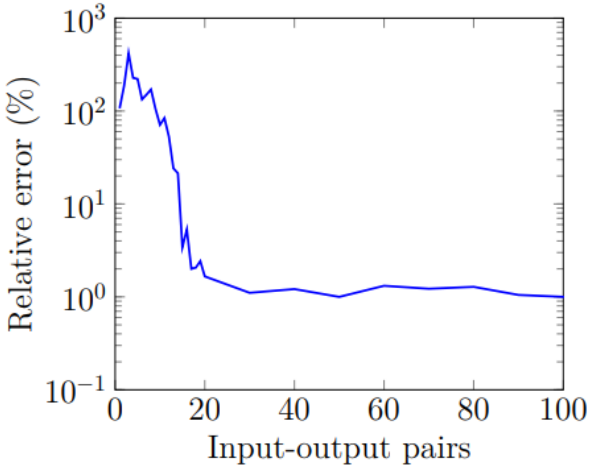
DeepGreen



[Gin et al., 2020]

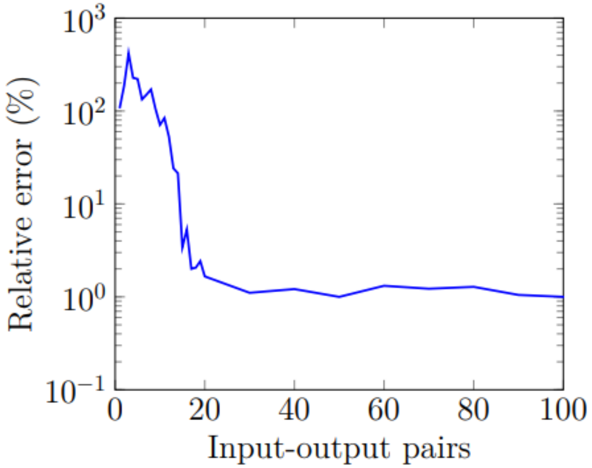
Main challenges of PDE learning

1. Theoretical results



Main challenges of PDE learning

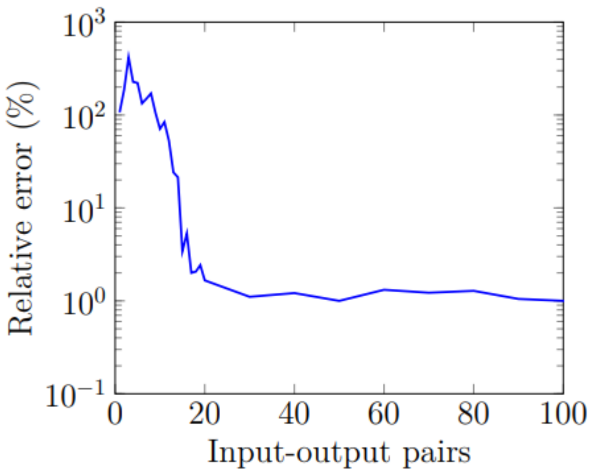
1. Theoretical results



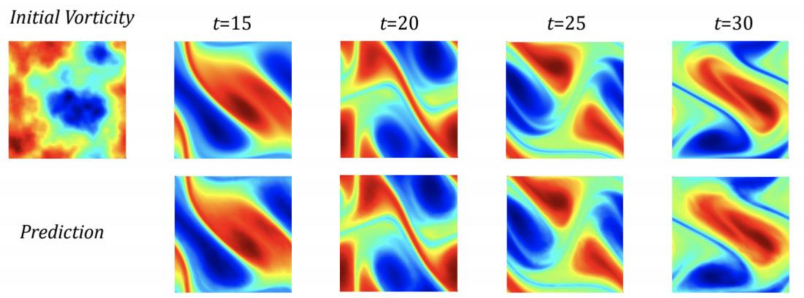
- Type and number of training data
- Performance guarantees
- Neural network architectures
- Noise robustness

Main challenges of PDE learning

1. Theoretical results



2. Interpretability of the model

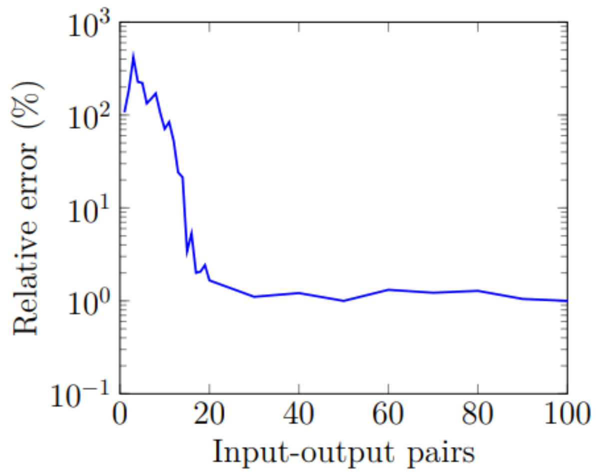


[Li et al, 2020]

- Type and number of training data
- Performance guarantees
- Neural network architectures
- Noise robustness

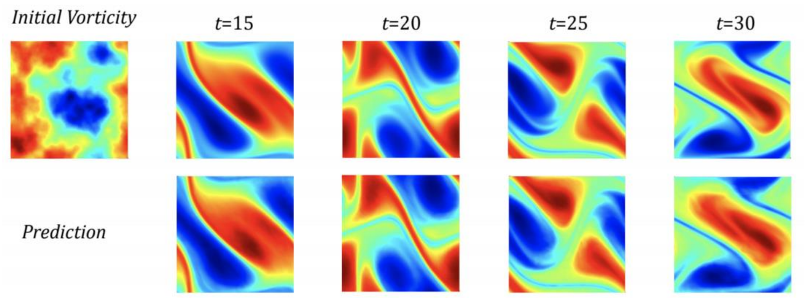
Main challenges of PDE learning

1. Theoretical results



- Type and number of training data
- Performance guarantees
- Neural network architectures
- Noise robustness

2. Interpretability of the model

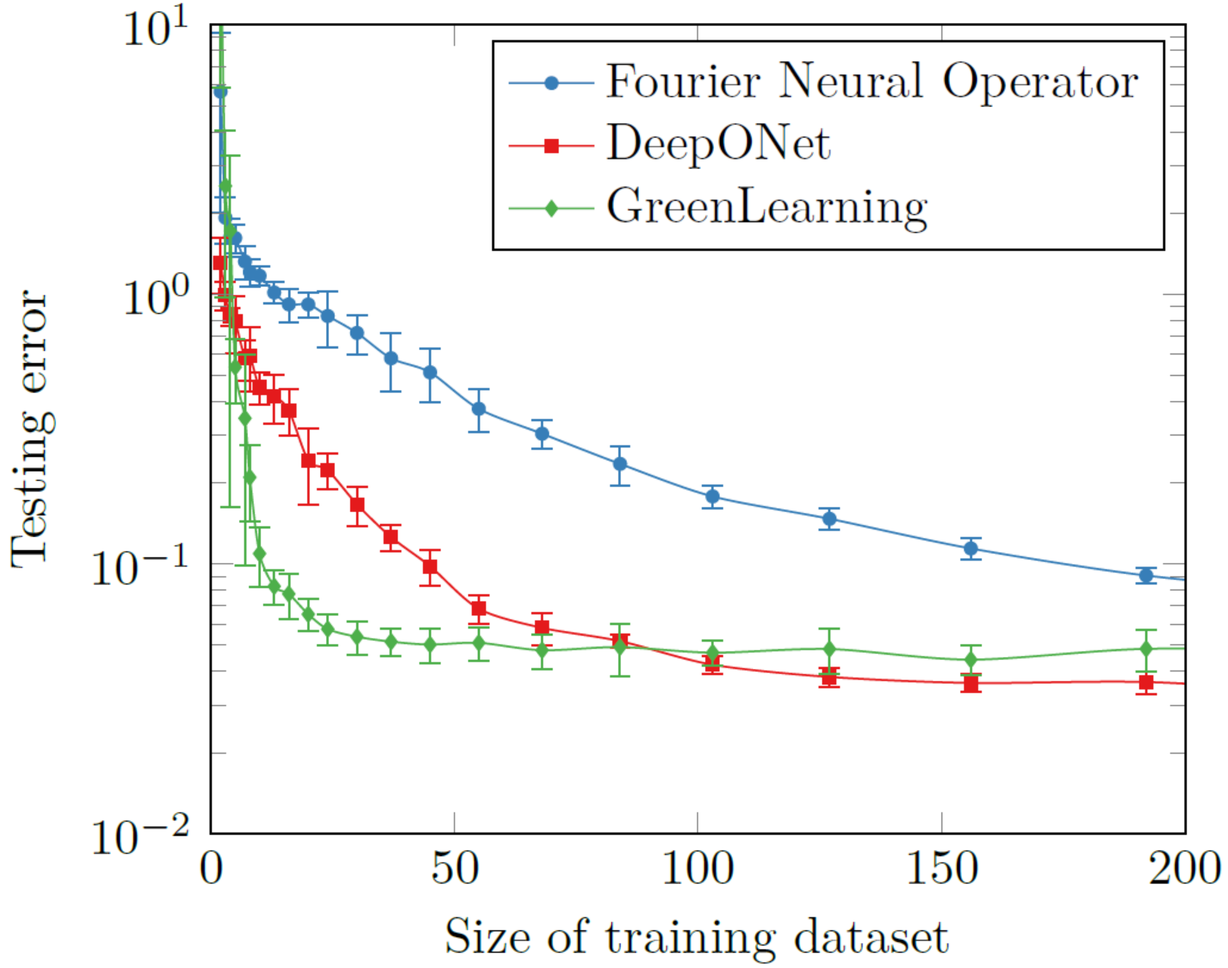


[Li et al, 2020]

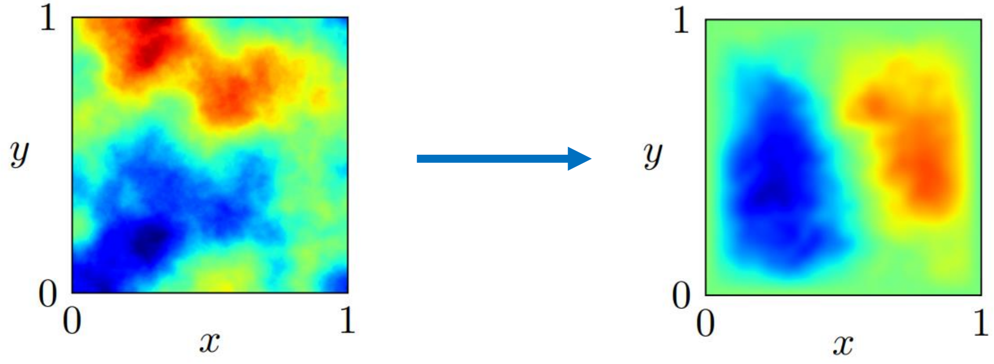
- Dominant modes
- Symmetries
- Conservation laws
- Singularities

PDE learning is data-efficient

Learn the solution operator of Poisson equation



Random source | Solution



How much training data is needed to learn a PDE?

<u>Related works:</u> [B., Townsend, 2022], [B. et al., 2022], [Chen et al., 2023], [de Hoop et al., 2021], [Lu et al., 2021], [Schäfer, Owhadi, 2021], [Schäfer et al., 2017]

Green's functions

Linear differential equation:

$$\mathcal{L}u = f \qquad u_{|\partial D} = 0 \quad \longrightarrow \quad u(x) = \int_D G(x,y) f(y)\, \mathrm{d}y$$

George Green

Green's functions

Linear differential equation:

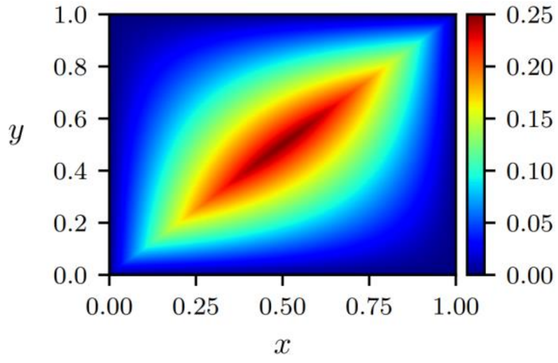
$$\mathcal{L}u = f$$
$$u_{|\partial D} = 0$$

$$\longrightarrow \quad u(x) = \int_D G(x,y) f(y)\,\mathrm{d}y$$

George Green

Poisson equation

$$-\nabla^2 u = f$$
$$u(0) = u(1) = 0$$



Green's functions

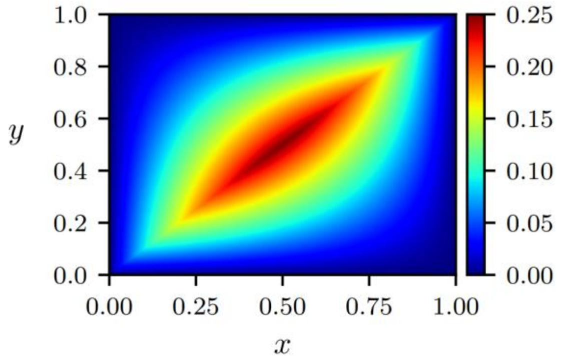
Linear differential equation:

$$\mathcal{L}u = f \qquad u_{|\partial D} = 0 \quad \longrightarrow \quad u(x) = \int_D G(x,y) f(y)\,\mathrm{d}y$$

George Green

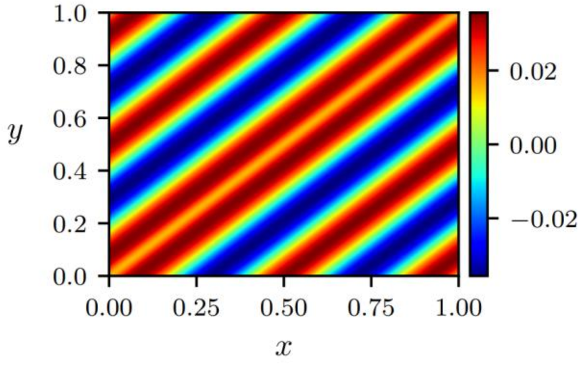
Poisson equation

$$-\nabla^2 u = f$$
$$u(0) = u(1) = 0$$



Helmholtz equation

$$\nabla^2 u + k^2 u = f$$
$$u(0) = u(1)$$



Learning Green's functions of elliptic PDEs

Elliptic PDEs in 3D of the form:

$$\mathcal{L}u := -\nabla \cdot (A(x)\nabla u) = f \quad \longrightarrow \quad u(x) = \int_D G(x,y) f(y)\,\mathrm{d}y$$

Theorem (B., Halikias, Townsend, 2023).

There is a randomized algorithm that achieves exponential convergence for learning the Green's function, with exceptionally high probability of success.

The proof combines core techniques in numerical analysis and generalizes them to infinite dimensions: randomized SVD + hierarchical matrices + peeling.

B., Halikias, Townsend, "Elliptic PDE learning is provably data-efficient", **PNAS**, 2023.

Randomized numerical linear algebra

Randomized singular value decomposition

Best rank k approximation

$$A = U \Sigma V^*$$

$$\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_k \geq \ldots$$

Eckart-Young theorem

$$\epsilon_k = \sqrt{\sum_{j=k+1}^{n} \sigma_j(A)^2}$$

Randomized singular value decomposition

Best rank k approximation

$$A = U \Sigma V^*$$

$$\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_k \geq \ldots$$

Eckart-Young theorem →

$$\epsilon_k = \sqrt{\sum_{j=k+1}^{n} \sigma_j(A)^2}$$

Theorem (Halko, Martinsson, Tropp, 2011).

We can construct an approximation A_k of A from k+5 random input vectors such that

$$\mathbb{P}\left[\|A - A_k\|_{\mathrm{F}} \leq (1 + 15\sqrt{k+5})\epsilon_k\right] \geq 0.999$$

Randomized singular value decomposition

$$A = U \; \Sigma \; V^*$$

$$\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_k \geq \ldots$$

Eckart-Young theorem →

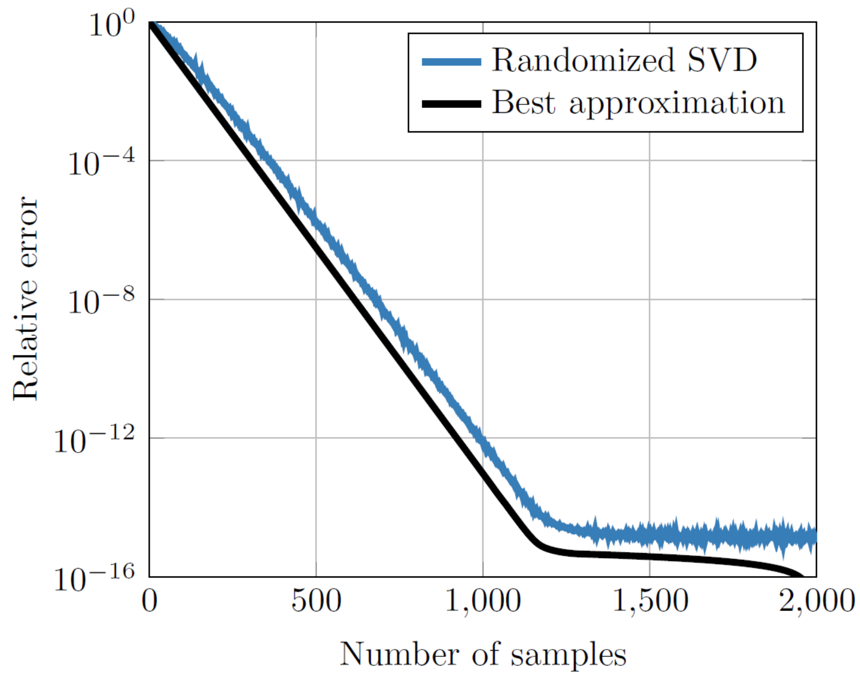
Best rank k approximation

$$\epsilon_k = \sqrt{\sum_{j=k+1}^{n} \sigma_j(A)^2}$$

Theorem (Halko, Martinsson, Tropp, 2011).

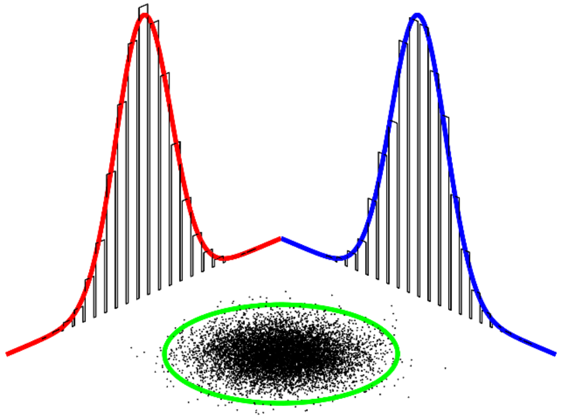
We can construct an approximation A_k of A from k+5 random input vectors such that

$$\mathbb{P}\left[\|A - A_k\|_{\mathrm{F}} \leq (1 + 15\sqrt{k+5})\epsilon_k\right] \geq 0.999$$



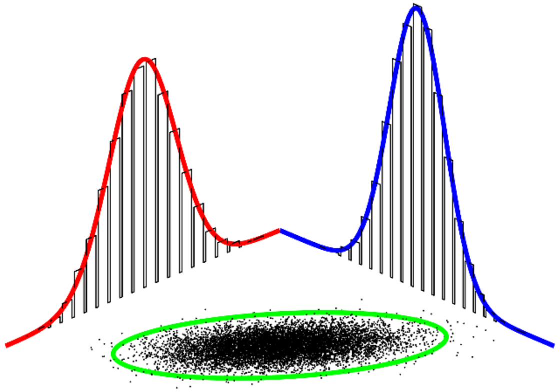
Generalization of the randomized SVD

Standard Gaussian vectors



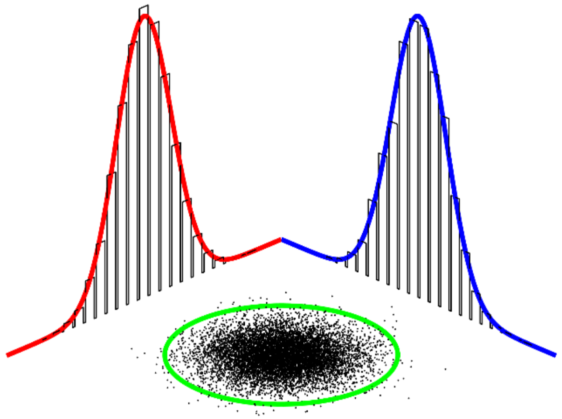
[B., Townsend, 2022]

Correlated Gaussian vectors



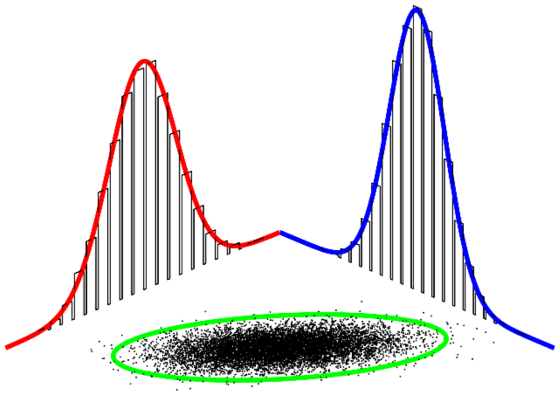
Generalization of the randomized SVD

Standard Gaussian vectors



[B., Townsend, 2022]

Correlated Gaussian vectors



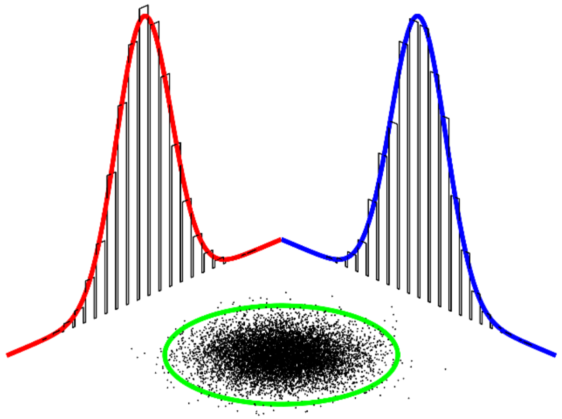
Theorem (B., Townsend, 2022).

We can construct an approximation A_k of A from k+5 correlated random input vectors such that

$$\mathbb{P}\left[\|A - A_k\|_{\mathrm{F}} \le \left(1+9\sqrt{k(k+5)\beta_k/\gamma_k}\right)\epsilon_k\right] \ge 0.999$$

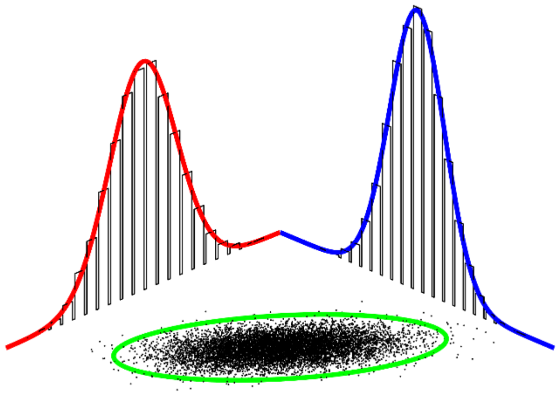
Generalization of the randomized SVD

Standard Gaussian vectors



[B., Townsend, 2022]

Correlated Gaussian vectors

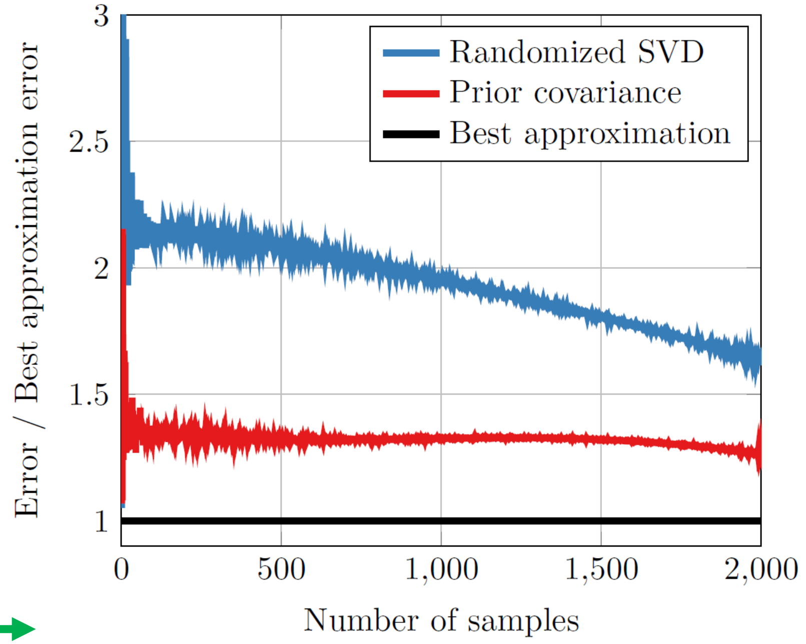


Theorem (B., Townsend, 2022).

We can construct an approximation A_k of A from k+5 correlated random input vectors such that

$$\mathbb{P}\left[\|A - A_k\|_{\mathrm{F}} \leq \left(1 + 9\sqrt{k(k+5)\beta_k/\gamma_k}\right)\epsilon_k\right] \geq 0.999$$

Prior knowledge on A



Hilbert-Schmidt operators

Definition:

Bounded linear operator $\mathscr{F}$ between Banach spaces with finite HS norm.

David Hilbert

Erhard Schmidt

Hilbert-Schmidt operators

Definition:

Bounded linear operator $\mathscr{F}$ between Banach spaces with finite HS norm.

David Hilbert

Erhard Schmidt

Matrices

$$\begin{pmatrix} & & \\ & A & \\ & & \end{pmatrix}$$

$$\begin{aligned} \|A\|_{\mathrm{HS}} &= \|A\|_{\mathrm{F}} \\ &= \sqrt{\mathrm{Tr}(A^* A)} \end{aligned}$$

Hilbert-Schmidt operators

Definition:

Bounded linear operator $\mathscr{F}$ between Banach spaces with finite HS norm.

David Hilbert

Erhard Schmidt

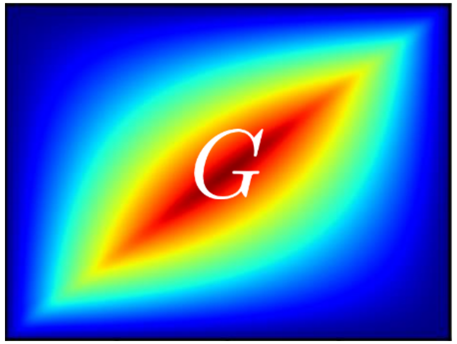
Matrices

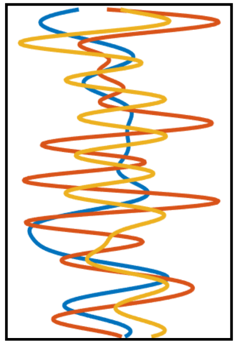
$$\begin{pmatrix} & & \\ & A & \\ & & \end{pmatrix}$$

$$\|A\|_{\mathrm{HS}} = \|A\|_{\mathrm{F}} = \sqrt{\mathrm{Tr}(A^* A)}$$

Integral operators

$$\mathscr{F}(f)(x) = \int_D G(x,y) f(y)\,\mathrm{d}y$$





Properties of Hilbert-Schmidt operators

Norm

$$\|\mathscr{F}\|_{\mathrm{HS}} = \|G\|_{L^2(D\times D)}$$

Properties of Hilbert-Schmidt operators

Norm

$$\|\mathscr{F}\|_{\mathrm{HS}} = \|G\|_{L^2(D\times D)}$$

Singular value decomposition

$$\mathscr{F}(f) = U \, \Sigma \, V^* \, f$$

$$\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_k \geq \ldots$$

Properties of Hilbert-Schmidt operators

Norm

$$\|\mathscr{F}\|_{\mathrm{HS}} = \|G\|_{L^2(D\times D)}$$

Singular value decomposition

$$\mathscr{F}(f) = U \, \Sigma \, V^* f$$

$$\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_k \geq \ldots$$

Eckart-Young-Mirsky theorem

Truncating the SVD gives the best rank k approximation in the HS norm:

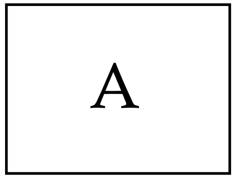
$$\epsilon_k = \|\mathscr{F} - \mathscr{F}_k\|_{\mathrm{HS}} = \sqrt{\sum_{j=k+1}^{\infty} \sigma_j^2}$$

Carl Eckart

Leon Mirsky

Randomized SVD for HS operators

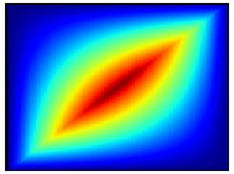
$$A \times F$$

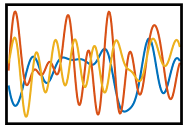


0.7	0.1
-0.4	0.2
0.8	0.5

[B., Townsend, 2022]

$$\int_D G(x,y) f(y)\, \mathrm{d}y$$





Randomized SVD for HS operators

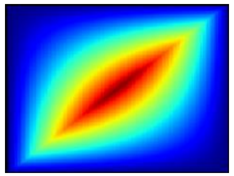
$$A \times F$$

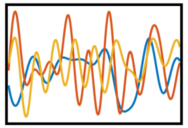
A

0.7	0.1
-0.4	0.2
0.8	0.5

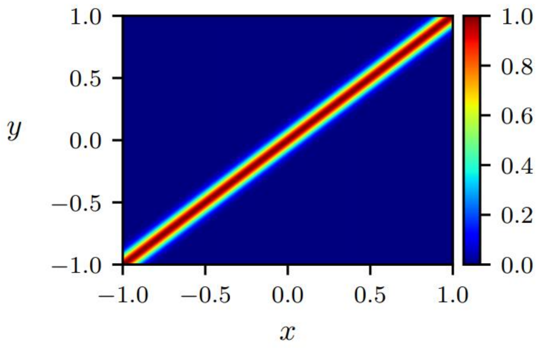
[B., Townsend, 2022]

$$\int_D G(x,y) f(y)\, \mathrm{d}y$$

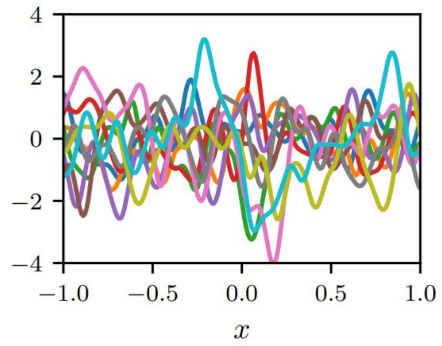




Covariance kernel

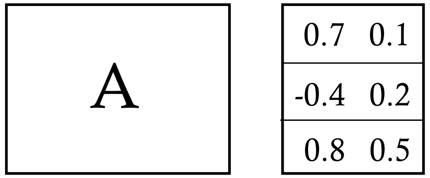


Random functions



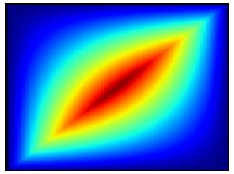
Randomized SVD for HS operators

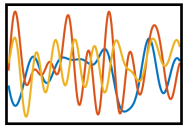
$$A \times F$$



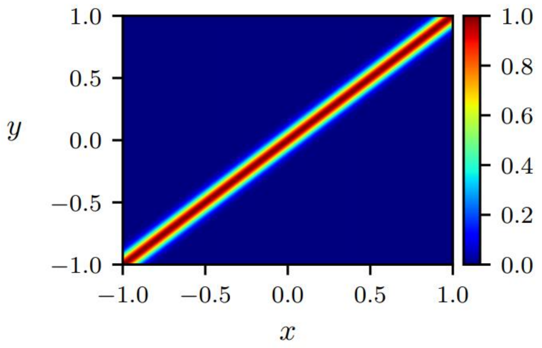
[B., Townsend, 2022]

$$\int_D G(x,y) f(y)\,\mathrm{d}y$$

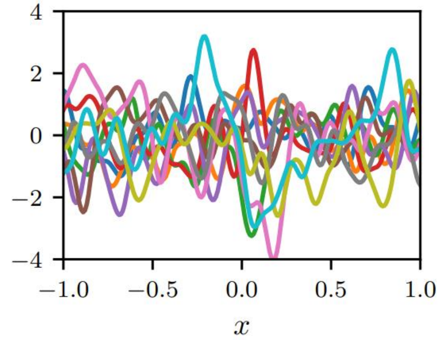




Covariance kernel



Random functions

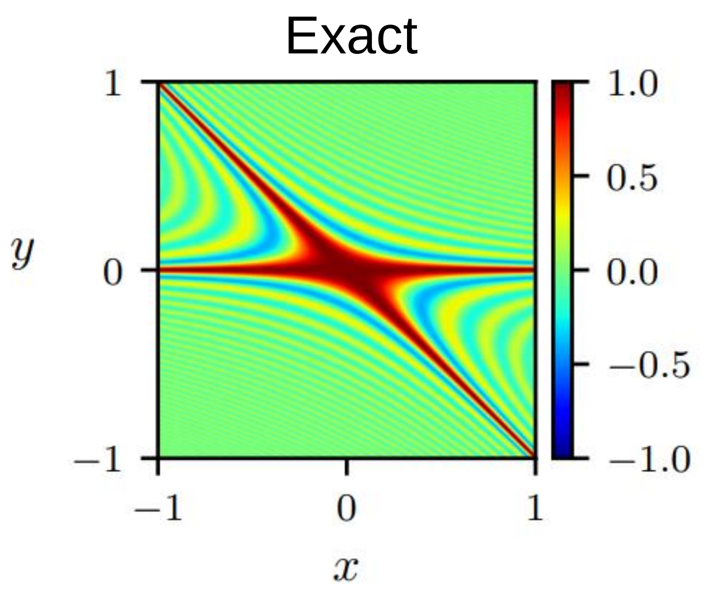


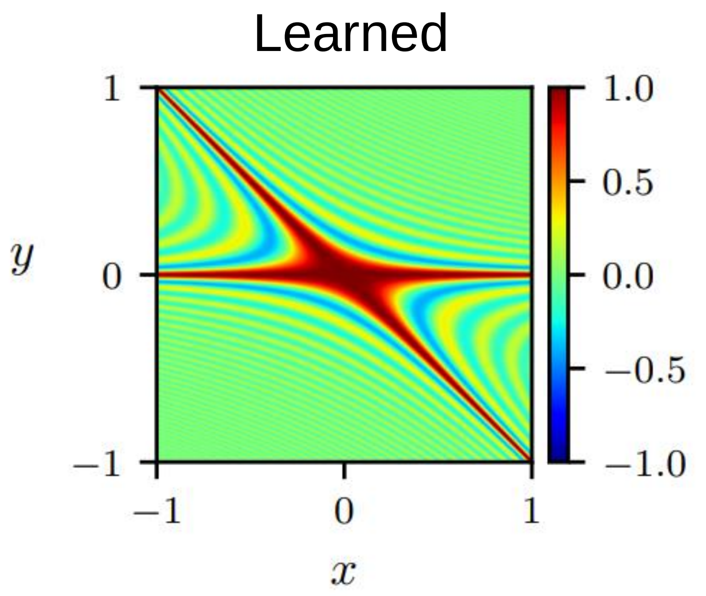
Theorem (B., Townsend, 2022).

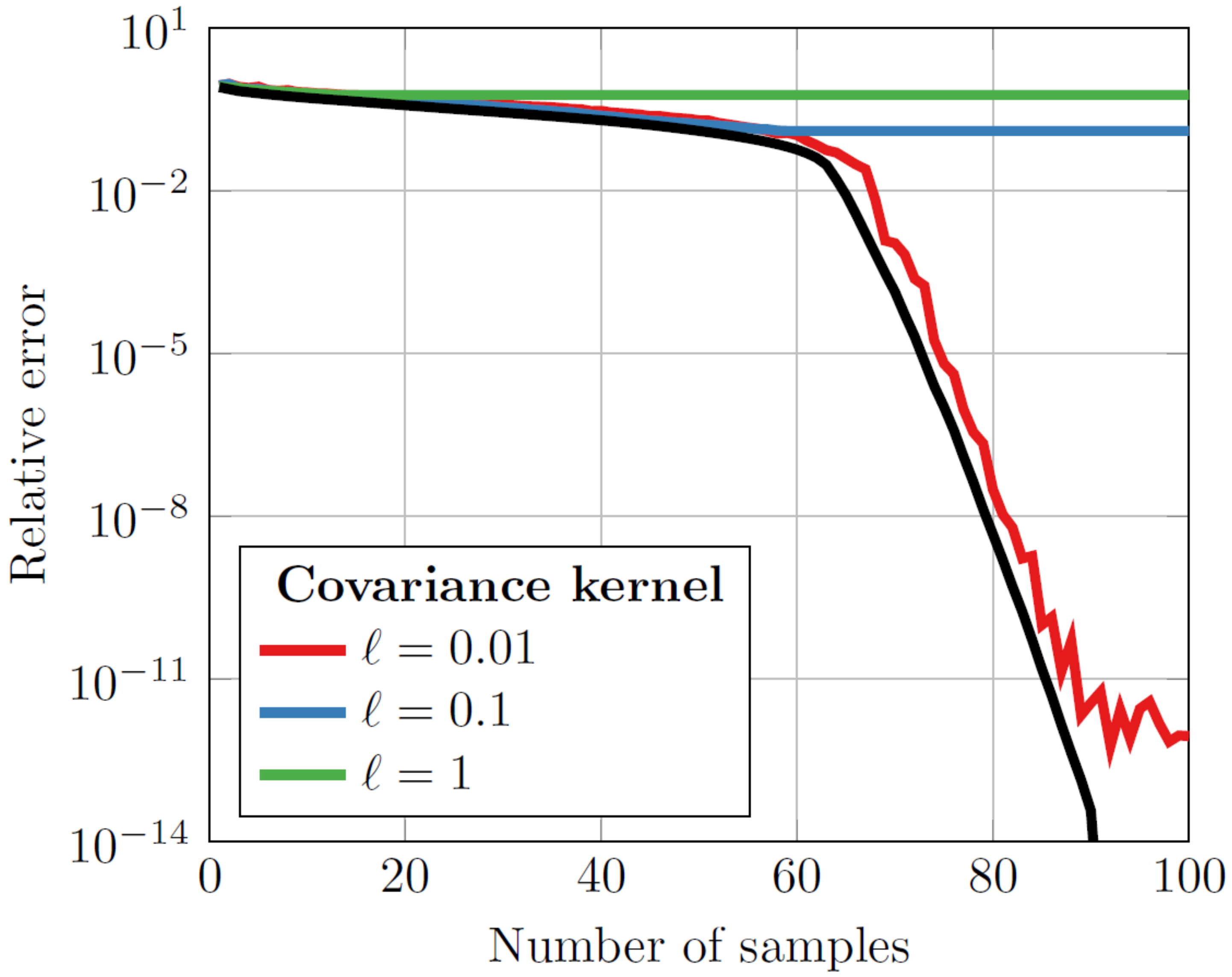
We can construct an approximation G_k of G from k+5 random input functions f such that

$$\mathbb{P}\left[\|G - G_k\|_{L^2} \leq \mathcal{O}\left(\sqrt{k^2/\gamma_k}\right)\epsilon_k\right] \geq 0.999$$

Randomized SVD for HS operators

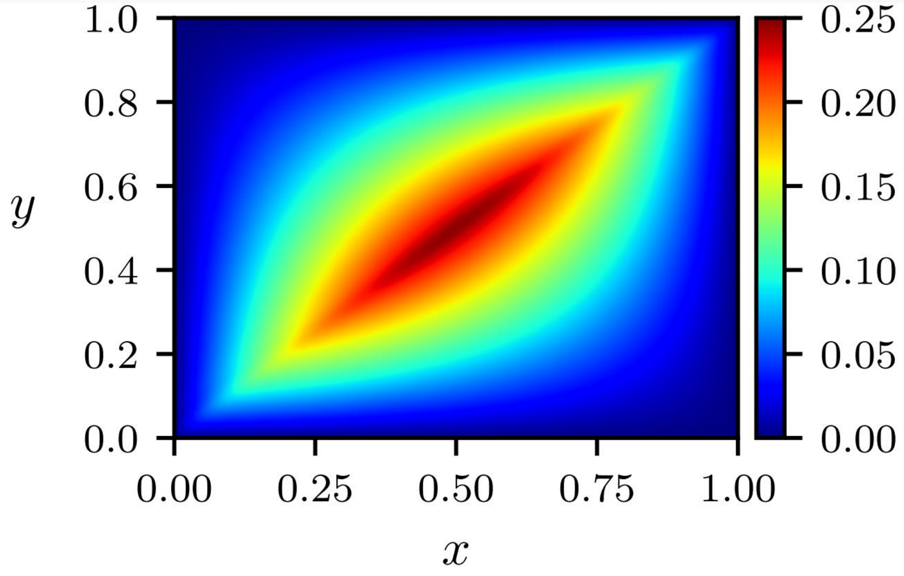






Randomized SVD for Green's functions

Green's function



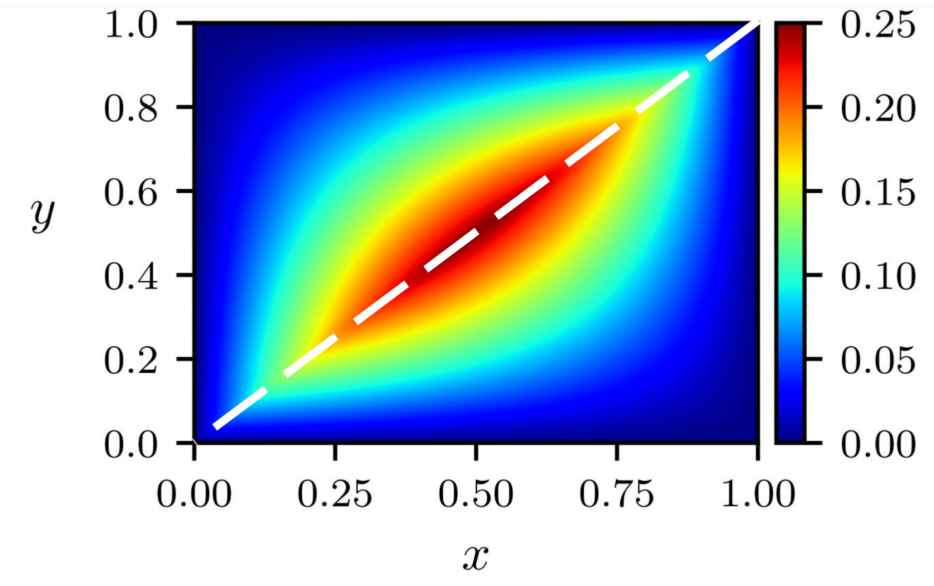
Theorem (B., Townsend, 2022).

We can construct an approximation G_k of G from k+5 random input functions f such that

$$\mathbb{P}\left[\|G - G_k\|_{L^2} \leq \mathcal{O}\left(\sqrt{k^2/\gamma_k}\right) \epsilon_k\right] \geq 0.999$$

Randomized SVD for Green's functions

Green's function



Theorem (B., Townsend, 2022).

We can construct an approximation G_k of G from k+5 random input functions f such that

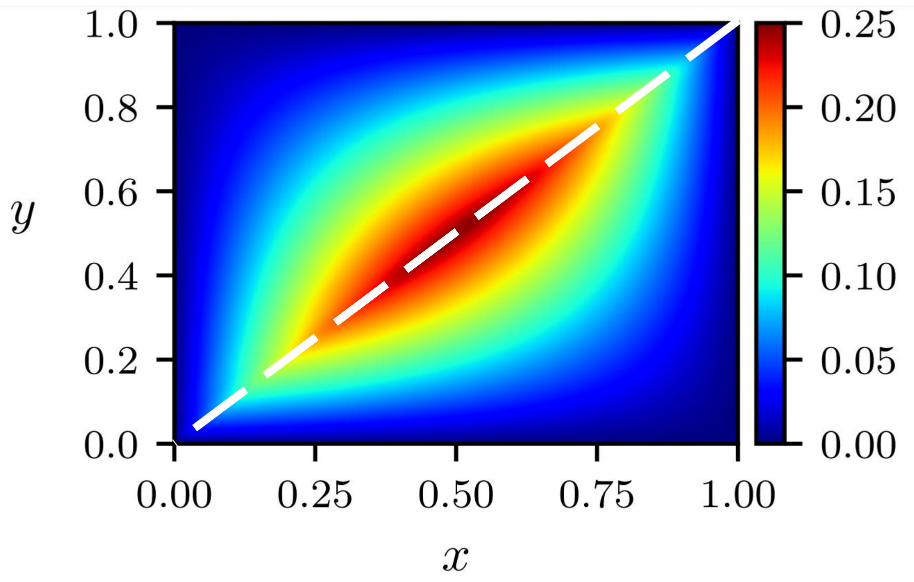
$$\mathbb{P}\left[\|G - G_k\|_{L^2} \leq \mathcal{O}\left(\sqrt{k^2/\gamma_k}\right)\epsilon_k\right] \geq 0.999$$

Problem:

The Green's functions are not smooth near the diagonal.

Randomized SVD for Green's functions

Green's function



Theorem (B., Townsend, 2022).

We can construct an approximation G_k of G from k+5 random input functions f such that

$$\mathbb{P}\left[\|G - G_k\|_{L^2} \leq \mathcal{O}\left(\sqrt{k^2/\gamma_k}\right) \epsilon_k\right] \geq 0.999$$

<u>Problem:</u>

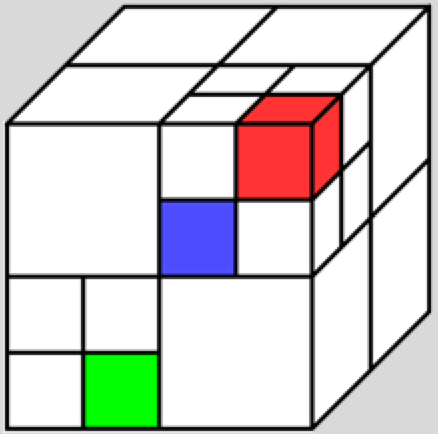
The Green's functions are not smooth near the diagonal.

→

ϵ_k decays very slowly with k

Regularity of the Green's function

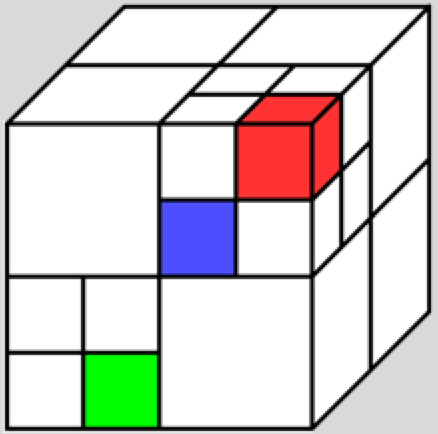
Low-rank structure



Low-rank structure on well separated domains.

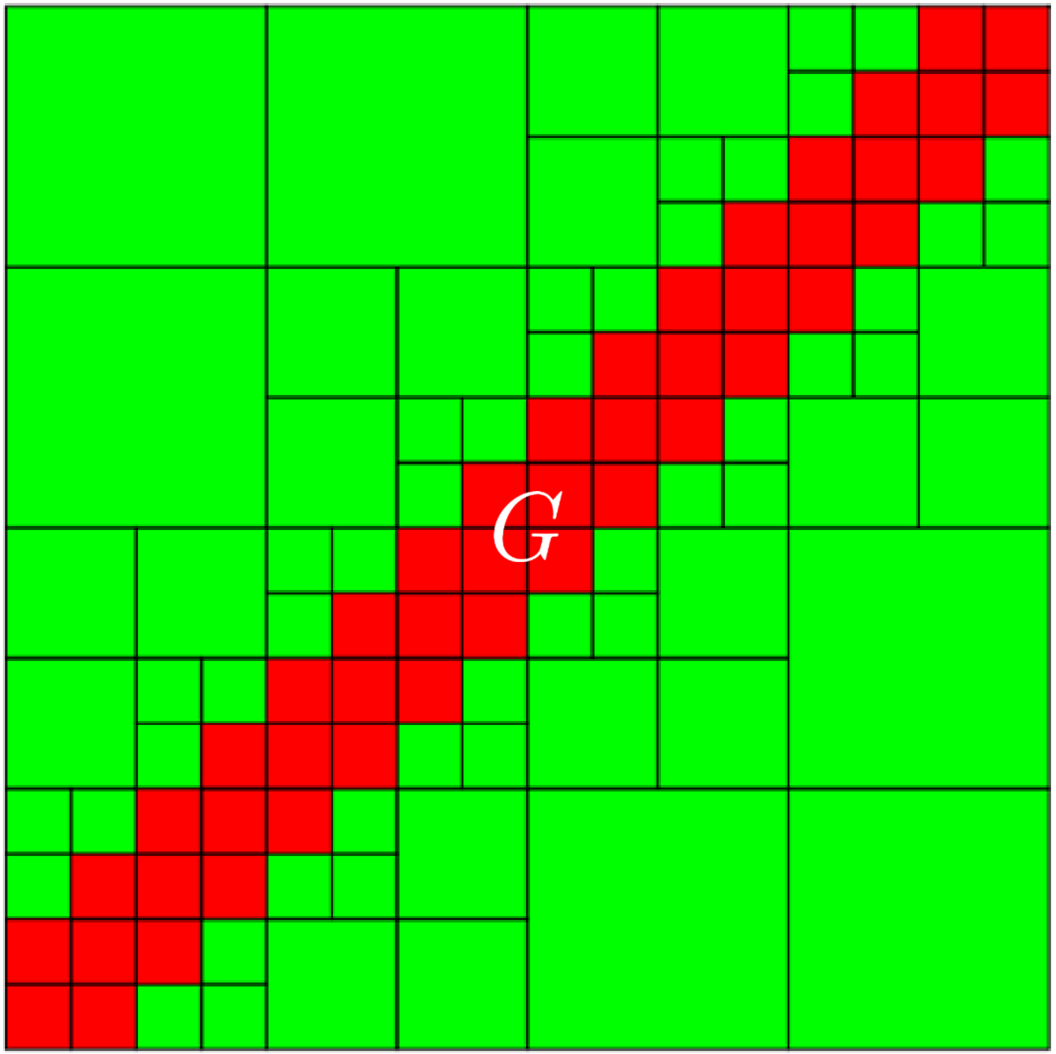
[Bebendorf, Hackbush, 2003]

Low-rank structure

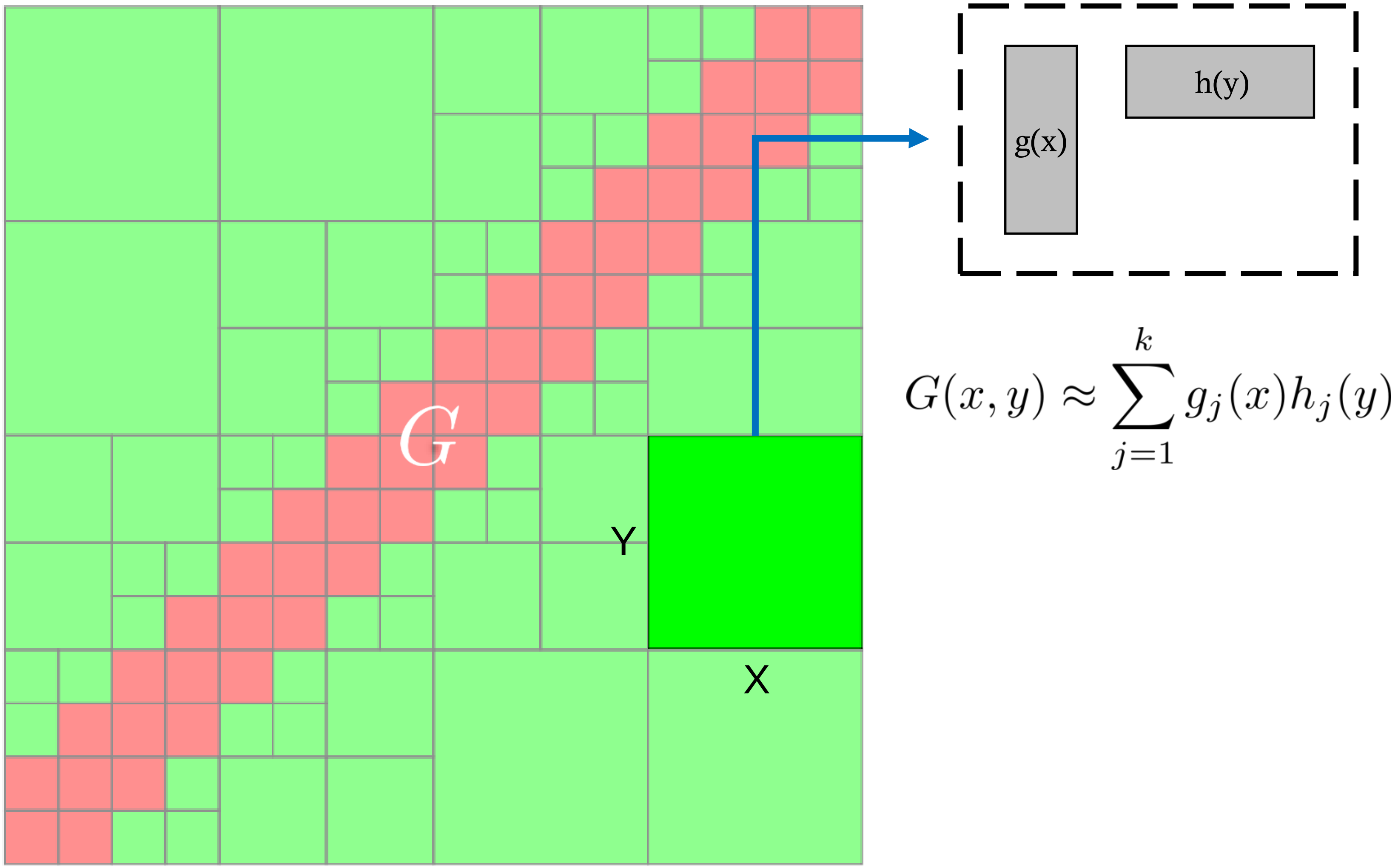


Low-rank structure on well separated domains.

[Bebendorf, Hackbush, 2003]



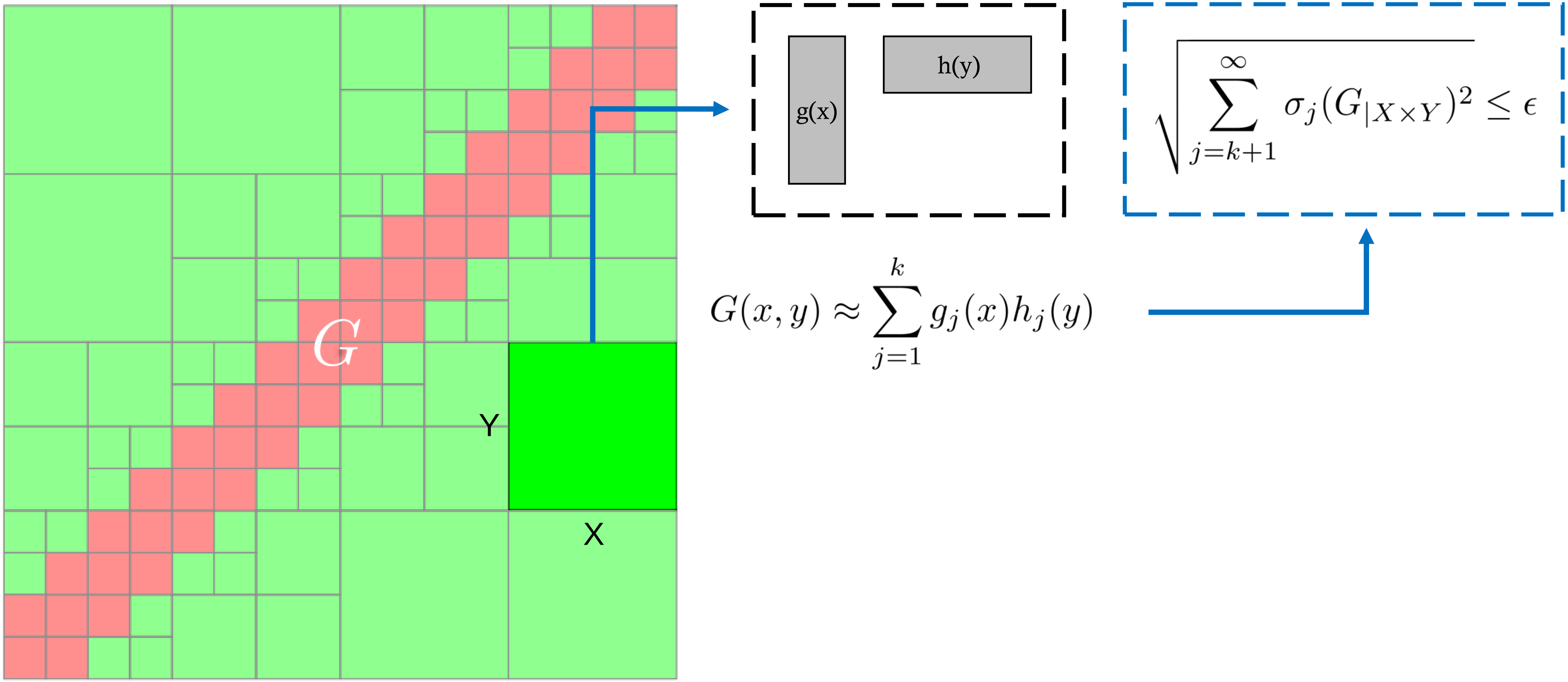
Hierarchical reconstruction



$$G(x,y) \approx \sum_{j=1}^{k} g_j(x) h_j(y)$$

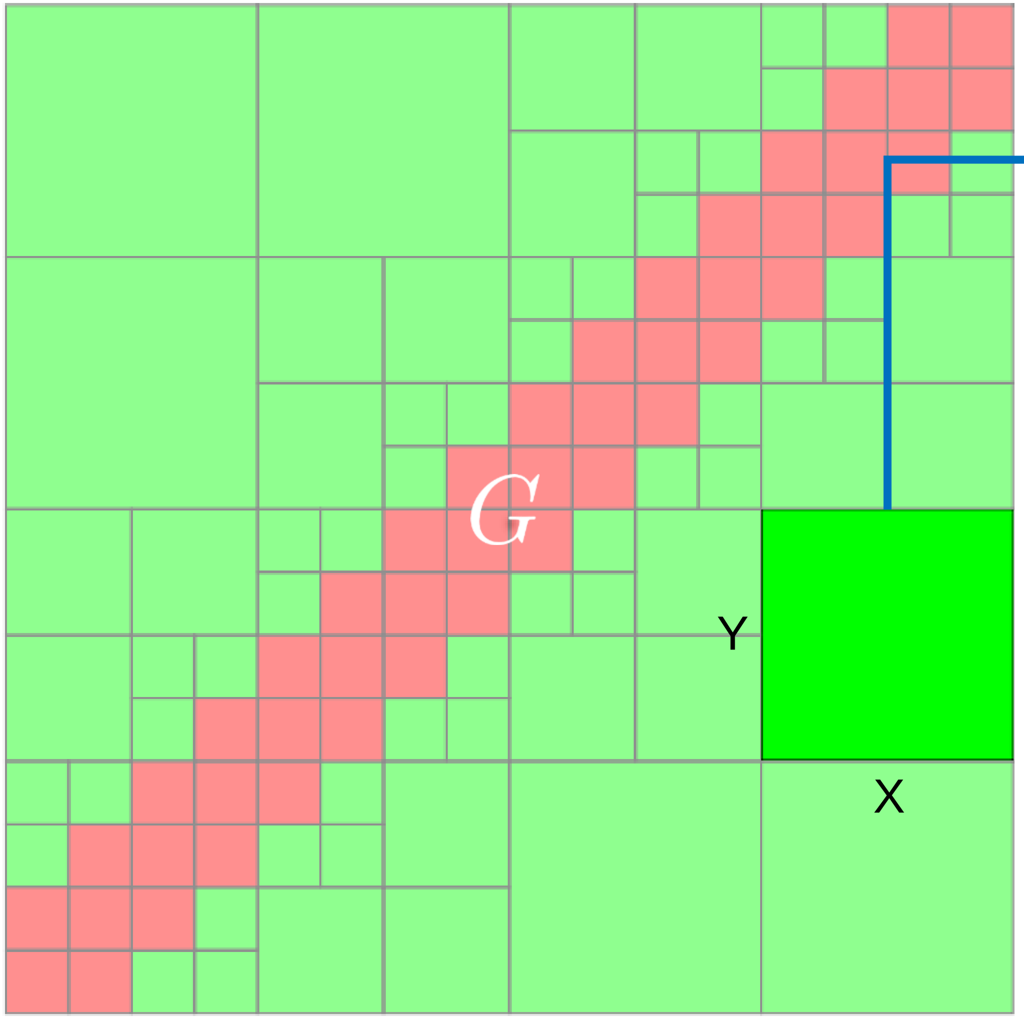
Hierarchical reconstruction

Randomized SVD on each subdomain

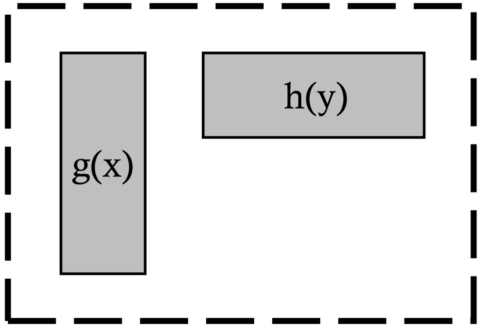


$$G(x,y) \approx \sum_{j=1}^{k} g_j(x) h_j(y)$$

Hierarchical reconstruction



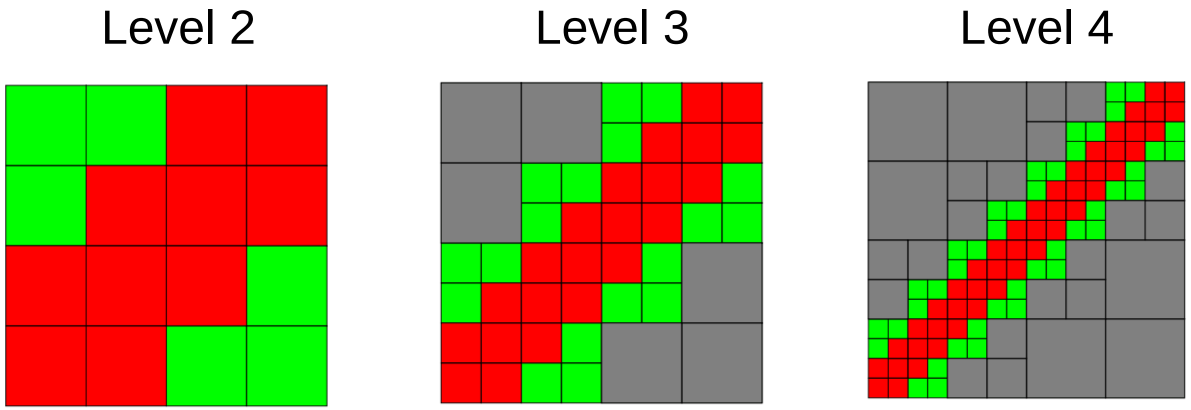
Randomized SVD on each subdomain



$$\sqrt{\sum_{j=k+1}^{\infty} \sigma_j(G_{|X\times Y})^2} \leq \epsilon$$

$$G(x,y) \approx \sum_{j=1}^{k} g_j(x)h_j(y)$$

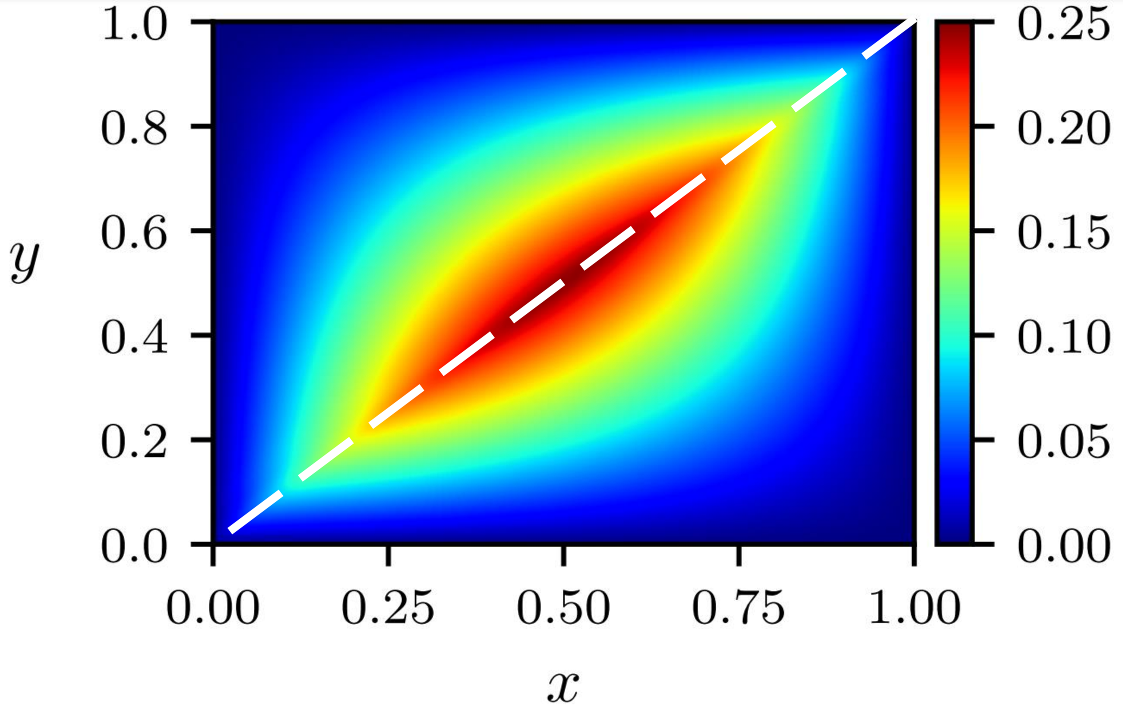
Hierarchical structure



Off-diagonal decay

Green's function of the Laplace operator:

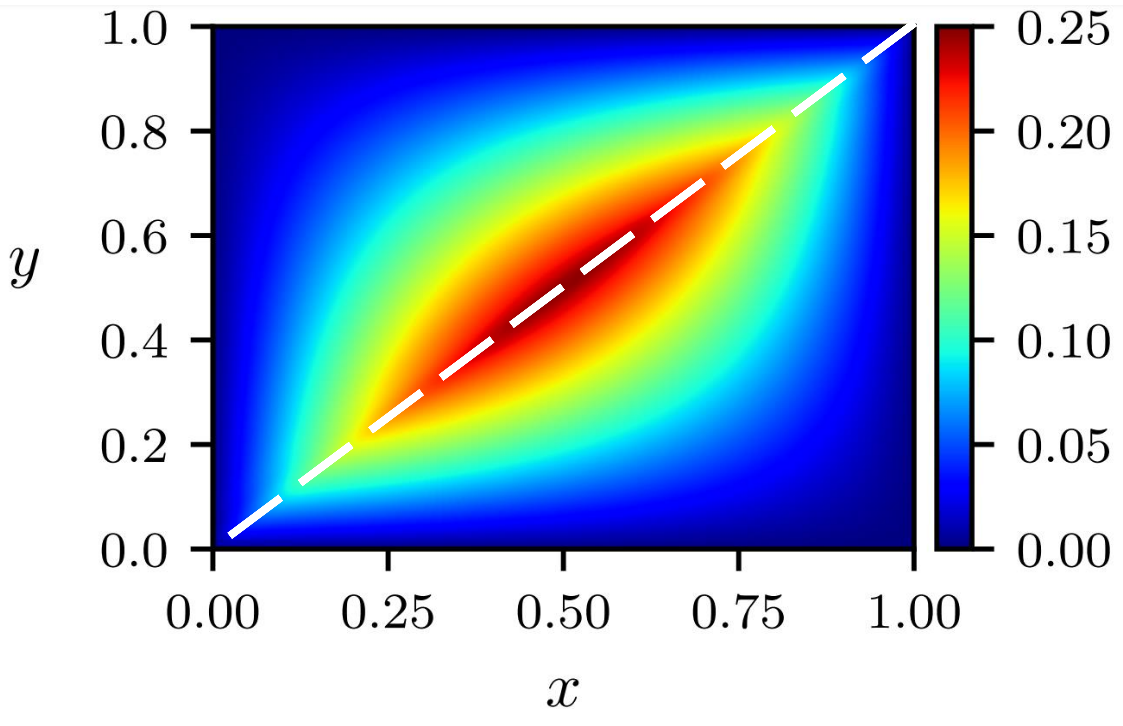
$$-\nabla^2 u = f$$



Off-diagonal decay

Green's function of the Laplace operator:

$$-\nabla^2 u = f$$



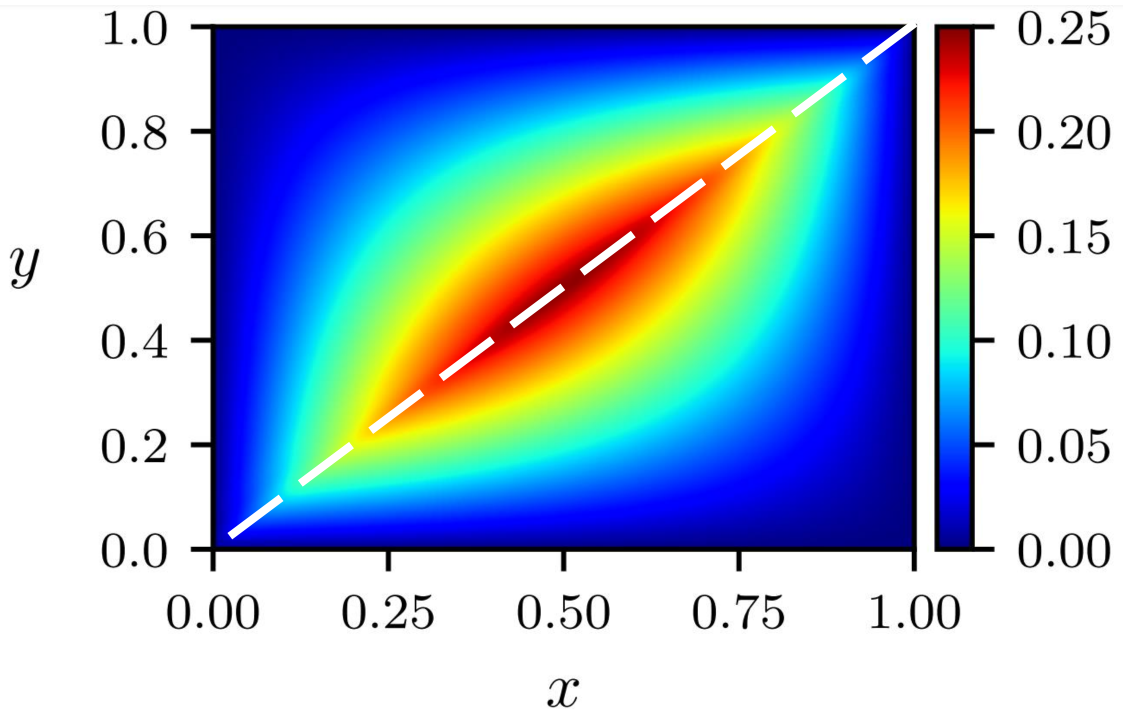
Green's functions are smooth and decay off the diagonal. [Grüter, Widman, 1982]

$$G(x,y) \leq \frac{1}{\|x-y\|}$$

Off-diagonal decay

Green's function of the Laplace operator:

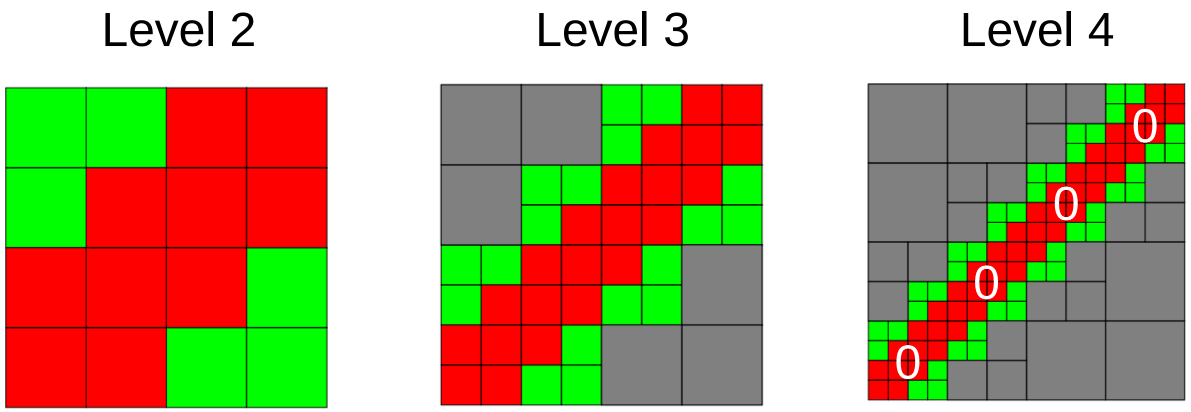
$$-\nabla^2 u = f$$



Green's functions are smooth and decay off the diagonal. [Grüter, Widman, 1982]

$$G(x,y) \leq \frac{1}{\|x-y\|}$$

Hierarchical structure



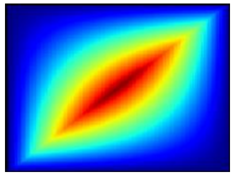
Summary of the result

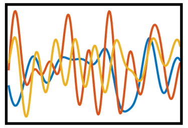
Theorem (B., Halikias, Townsend, 2023).

There is a randomized algorithm that achieves exponential convergence for learning the Green's function, with exceptionally high probability of success.

Randomized SVD

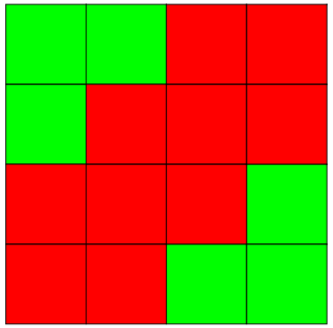
$$\int_D G(x,y)f(y)\,\mathrm{d}y$$



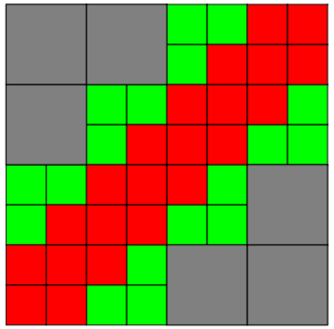


Hierarchical structure

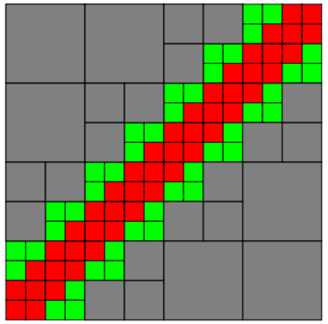
Level 2



Level 3



Level 4



Summary of the result

Theorem (B., Halikias, Townsend, 2023).

There is a randomized algorithm that achieves exponential convergence for learning the Green's function, with exceptionally high probability of success.

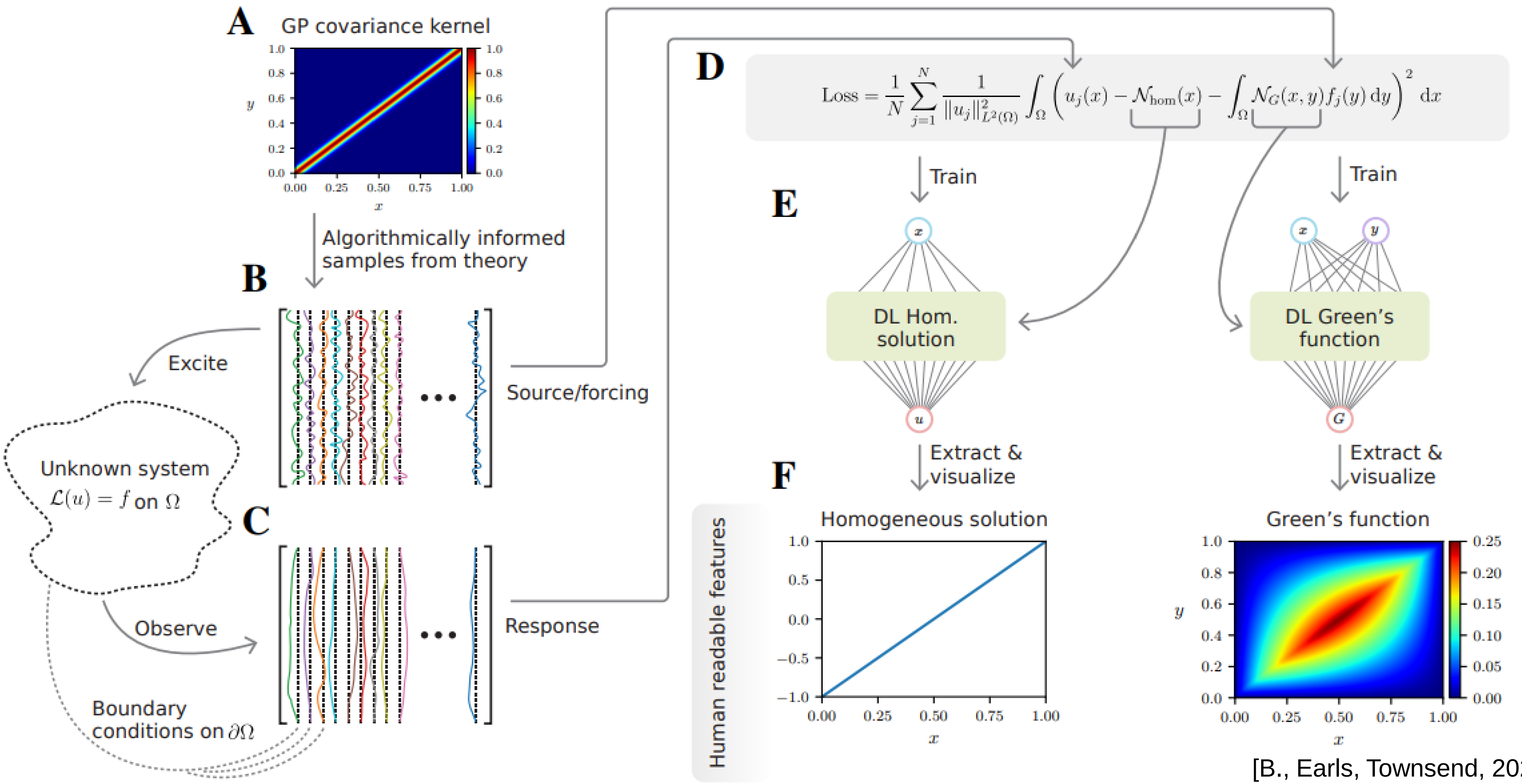
Extension to time-dependent PDEs of the form:

$$u_t - \nabla \cdot (A(x,t)\nabla u) = f(x,t)$$

B., Kim, Shi, Townsend, ''Learning Green's functions associated with time-dependent partial differential equations'', **J. Mach. Learn. Res.**, 2022.

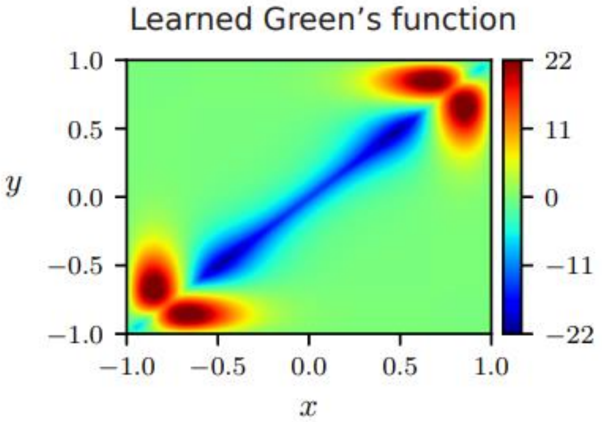
Deep learning applications

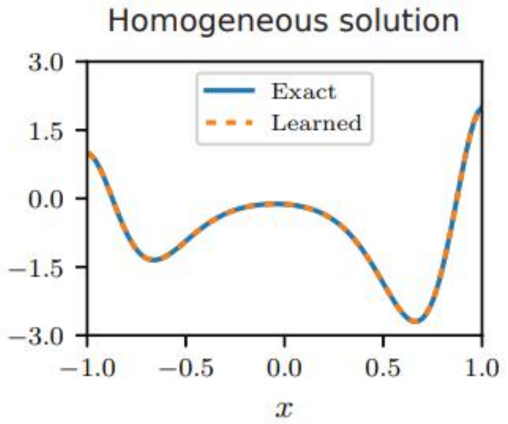
Deep learning method



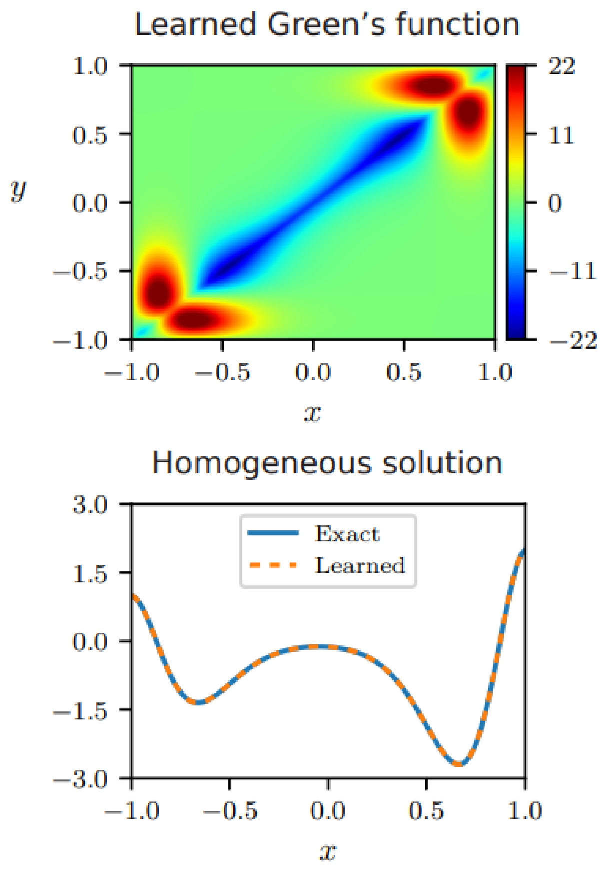
[B., Earls, Townsend, 2022]

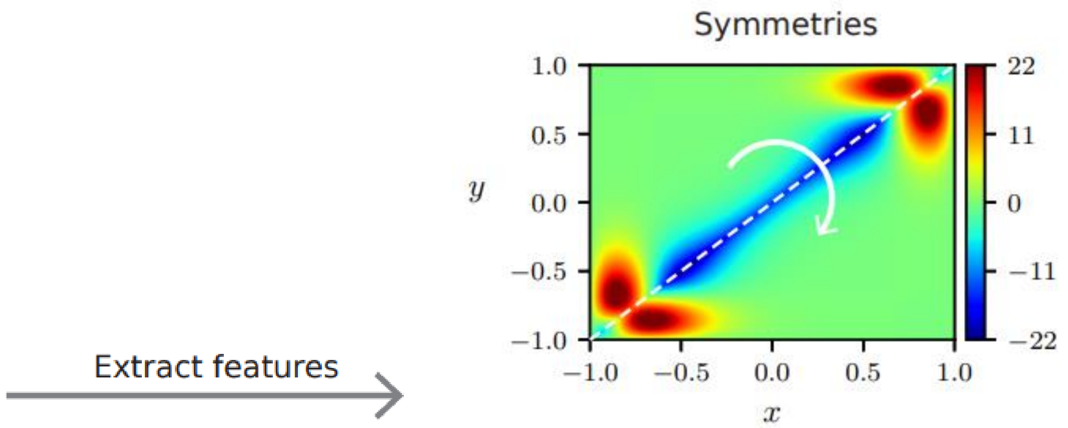
Learning features of the PDE



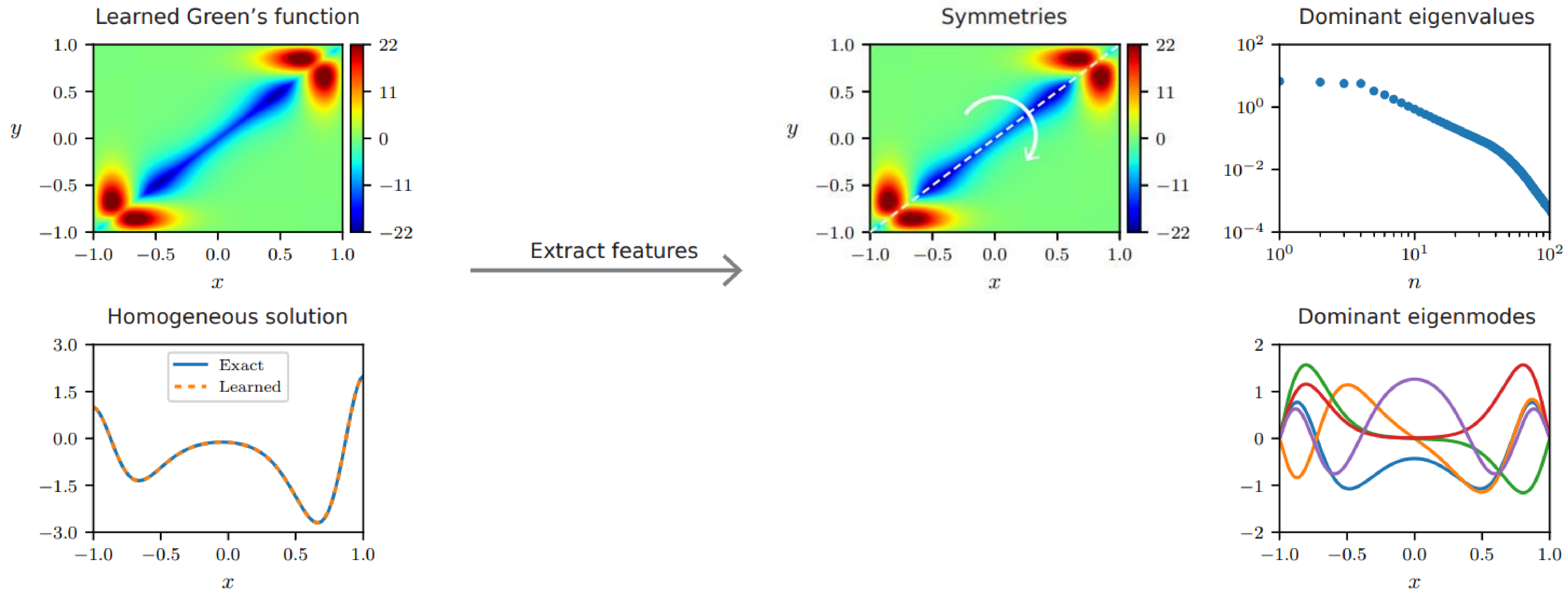


Learning features of the PDE

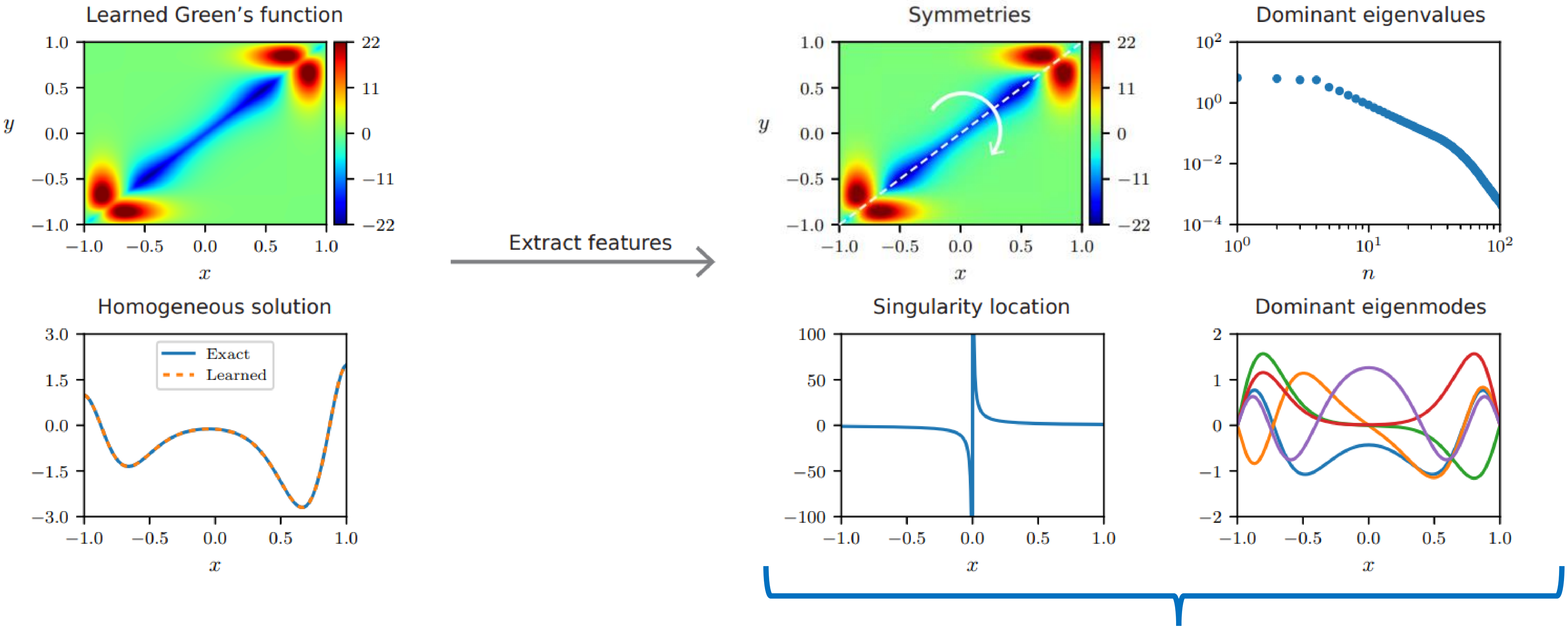




Learning features of the PDE



Learning features of the PDE



Rational neural networks have high approximation power and support feature extraction [B., Nakatsukasa, Townsend, 2020]

Advection-diffusion equation

Equation:

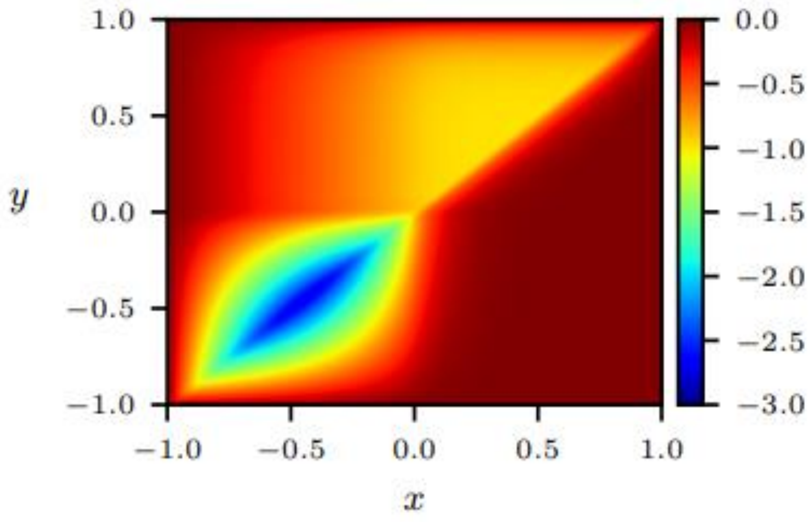
$$\mathcal{L}u = 0.1\frac{d^2u}{dx^2} + (x \geq 0)\frac{du}{dx}, \qquad u(-1) = 2, \quad u(1) = -1$$

Advection-diffusion equation

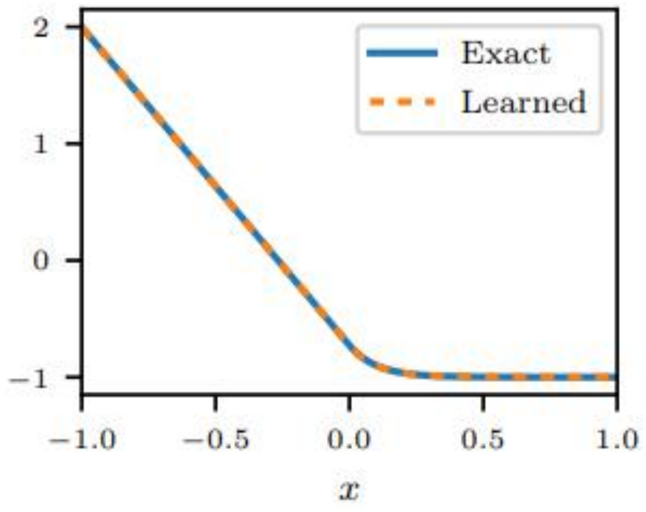
Equation:

$$\mathcal{L}u = 0.1\frac{d^2u}{dx^2} + (x \geq 0)\frac{du}{dx}, \qquad u(-1) = 2, \quad u(1) = -1$$

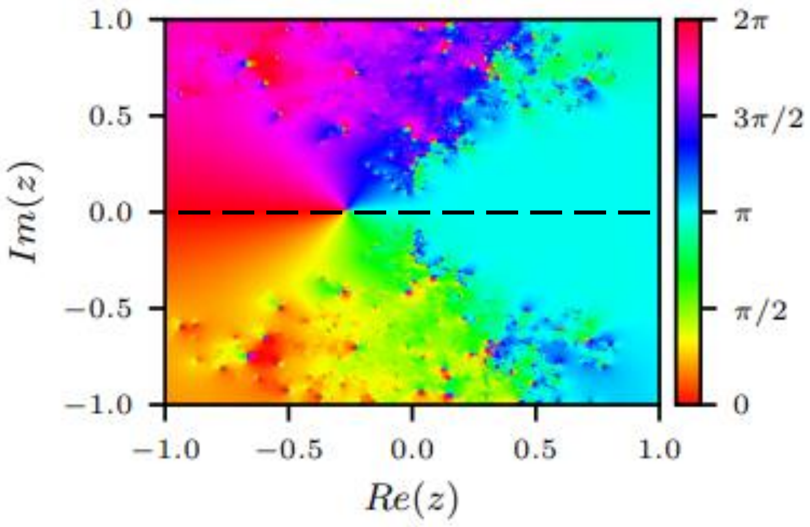
Green's function



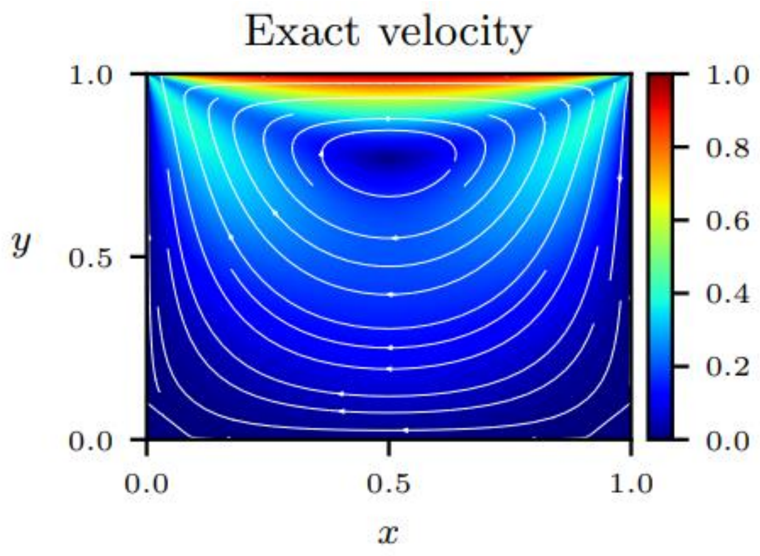
Homogeneous solution

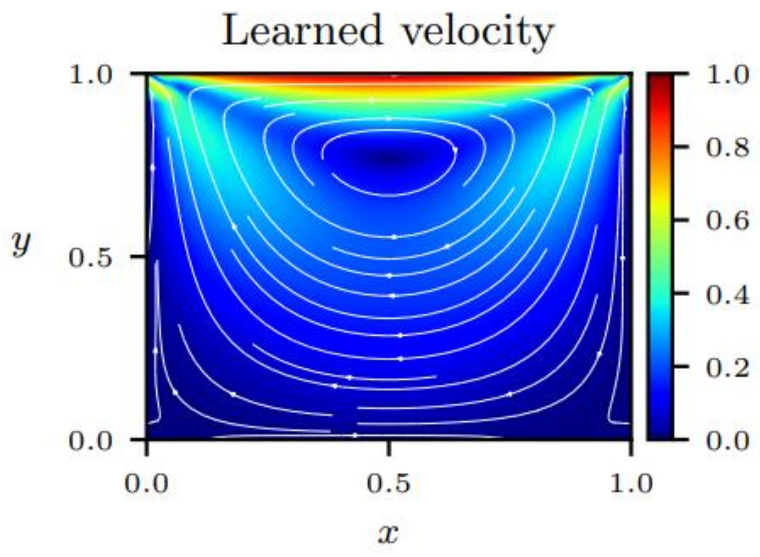


Phase portrait

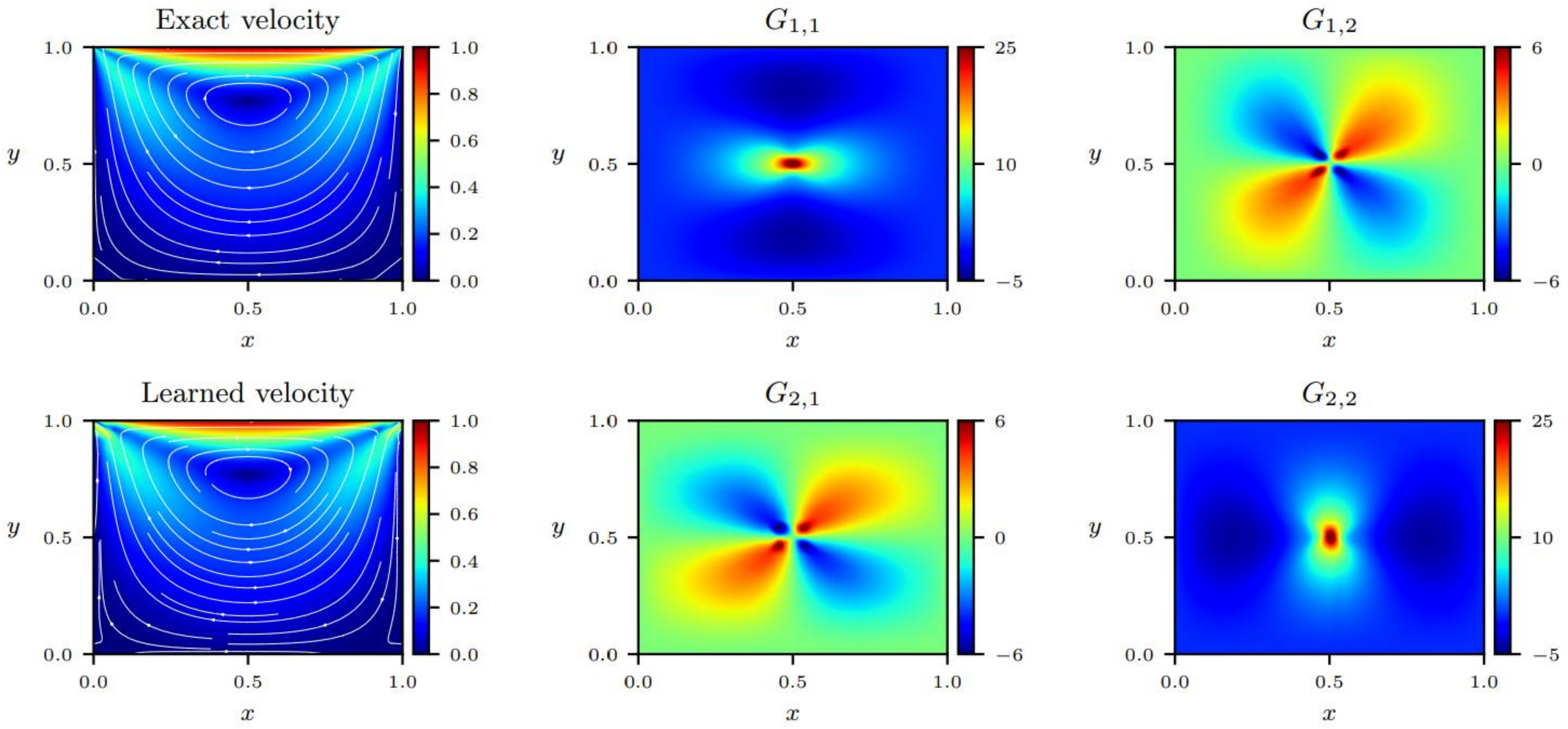


Stokes flow in a lid-driven cavity





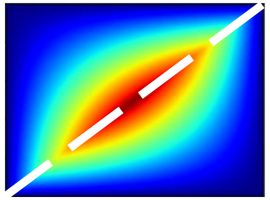
Stokes flow in a lid-driven cavity

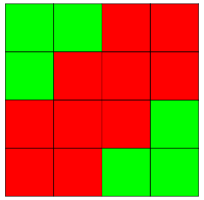


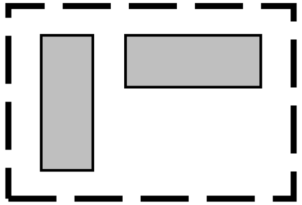
Conclusions

1. Theory for learning Green's functions

$$\mathcal{L}u = -\nabla \cdot (A(x)\nabla u)$$



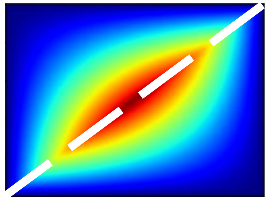
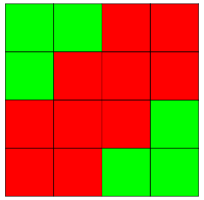
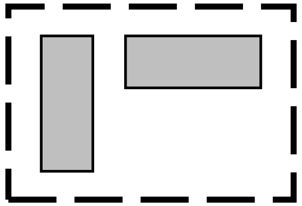




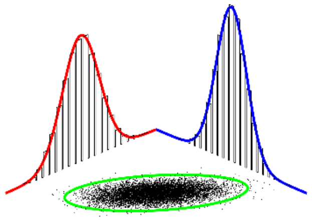
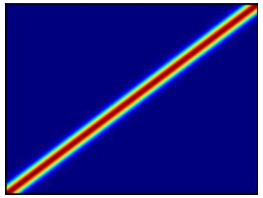
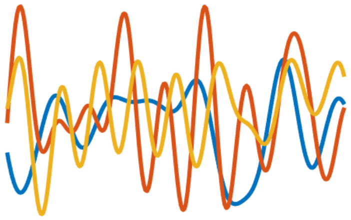
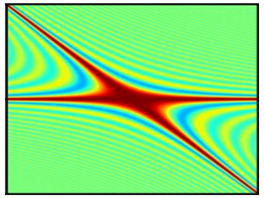
Conclusions

1. Theory for learning Green's functions

$$\mathcal{L}u = -\nabla \cdot (A(x)\nabla u)$$

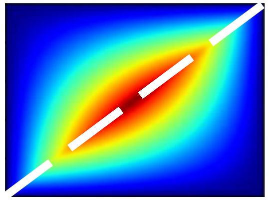
2. Generalization of the randomized SVD

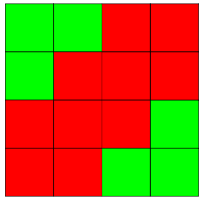





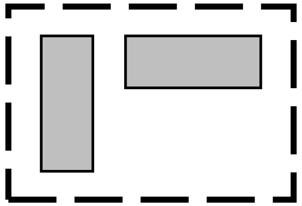
Conclusions

1. Theory for learning Green's functions

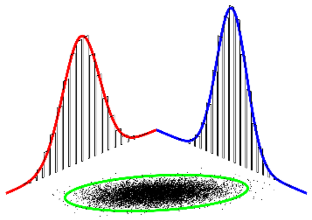
$$\mathcal{L}u = -\nabla \cdot (A(x)\nabla u)$$

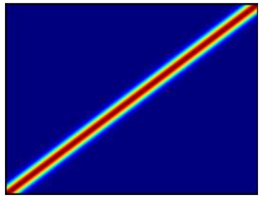


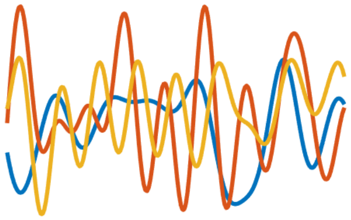


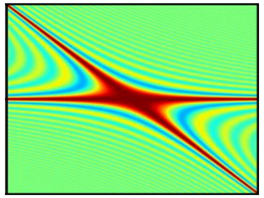


2. Generalization of the randomized SVD









3. Deep learning approach

TensorFlow

Python package

`pip install greenlearning`