Machine Learning for Many-Body Systems in Physics PART 2 !

CASUS
CENTER FOR ADVANCED SYSTEMS UNDERSTANDING
www.casus.science

Michael Hecht

University Kiel, 2022

HZDR HELMHOLTZ ZENTRUM DRESDEN ROSSENDORF
TECHNISCHE UNIVERSITÄT DRESDEN
CBG Max Planck Institute of Molecular Cell Biology and Genetics
UFZ HELMHOLTZ Centre for Environmental Research
Uniwersytet Wrocławski

SPONSORED BY THE
Federal Ministry of Education and Research
STAATSMINISTERIUM FÜR WISSENSCHAFT KULTUR UND TOURISMUS
Freistaat SACHSEN

The Bottleneck

How to compute Multivariate Function Expansions that closely approximate the ground truth function ?

- **Geometric approximation rates of Fourier expansions for analytic periodic functions, FFTs**

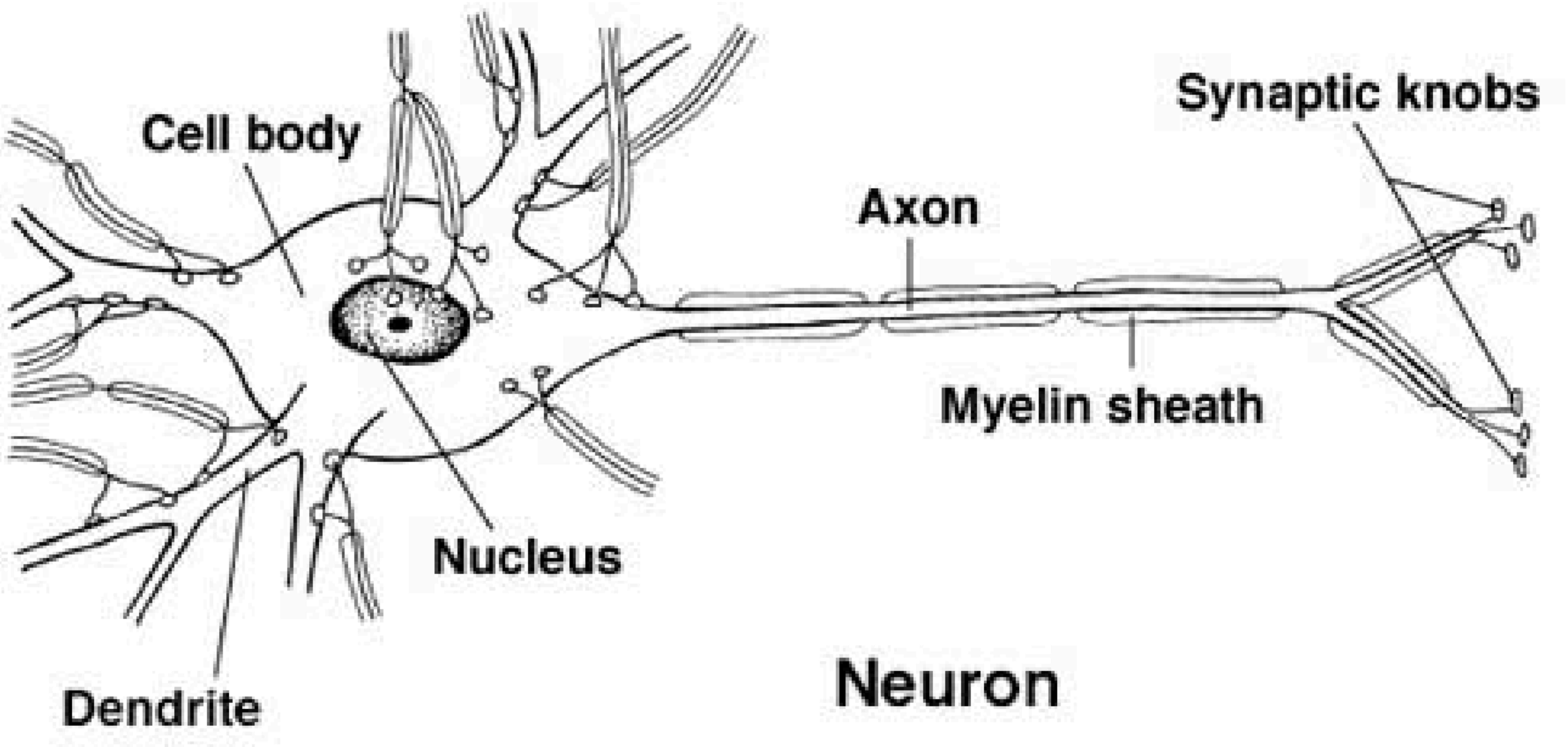
$$\theta(z) \approx \theta_n = \sum_{\|\alpha\|_\infty \leq n} c_\alpha e^{2\pi i \alpha \cdot z}, \qquad \|\theta - \theta_n\|_{C^0(\Omega)} = \mathcal{O}(r^{-n}), \qquad r > 1$$

- **Geometric approximation rates of the Coulomb / Gravitation field expansion in FMMs**

$$\phi(z) \approx \phi_n = Q \log(z) + \sum_{k=1}^{n} \frac{a_k}{z_k}, \qquad \|\phi - \phi_n\|_{C^0(\Omega)} = \mathcal{O}(z^{-n}), \qquad |z| > 1\,.$$

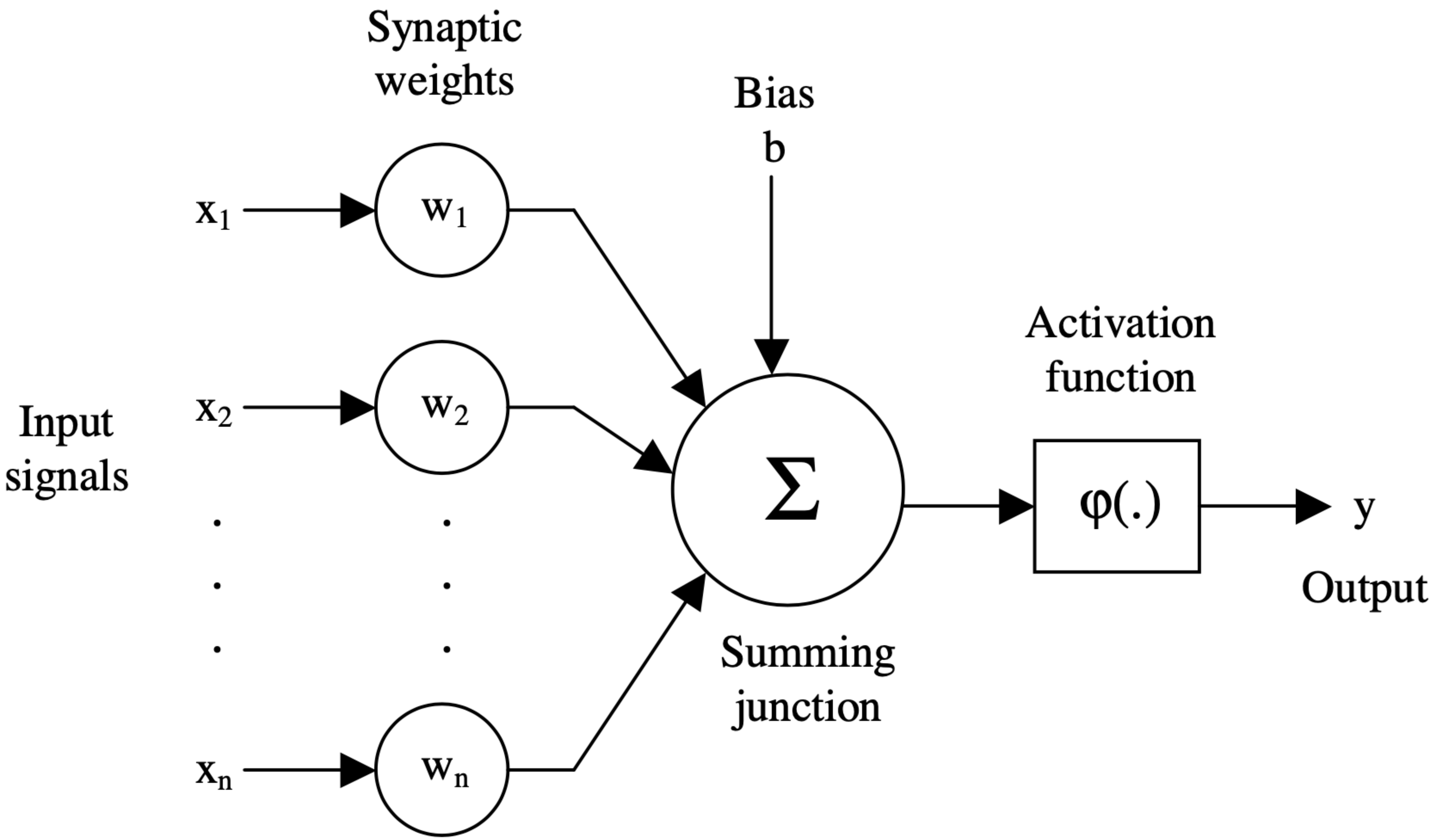
Gauss, C. F. (1886). Theoria interpolationis methodo nova tractata Werke band 3, 265–327. *Göttingen: Königliche Gesellschaft der Wissenschaften*.
Cooley, J. W., & Tukey, J. W. (1965). An algorithm for the machine calculation of complex Fourier series. *Mathematics of computation*, *19*(90), 297-301.
Greengard, L., & Rokhlin, V. (1987). A fast algorithm for particle simulations. *Journal of computational physics*, *73*(2), 325-348.

What are Neural Nets (NNs) ?



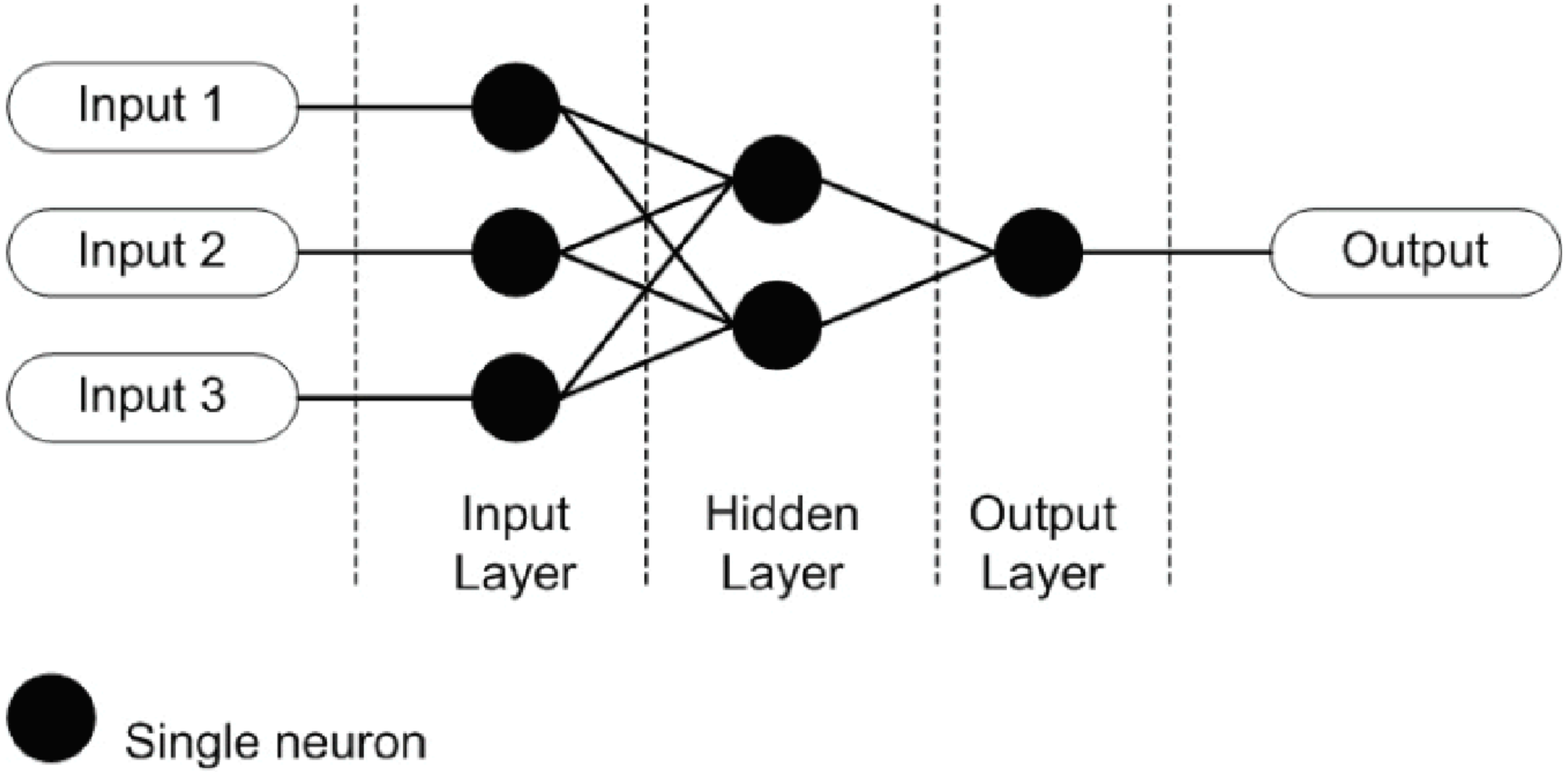
Csáji, B. C. (2001). Approximation with artificial neural networks. *Faculty of Sciences, Etvs Lornd University, Hungary*, *24*(48), 7.
Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT press.

What are Neural Nets (NNs) ?



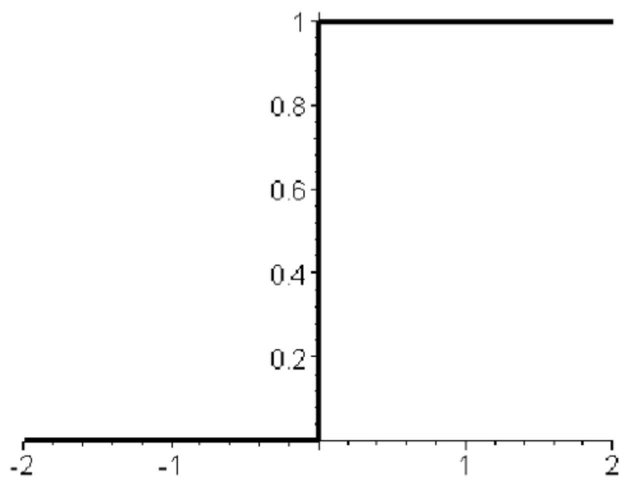
Csáji, B. C. (2001). Approximation with artificial neural networks. *Faculty of Sciences, Etvs Lornd University, Hungary*, *24*(48), 7.
Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT press.

Combination of the Neurons yields NNs



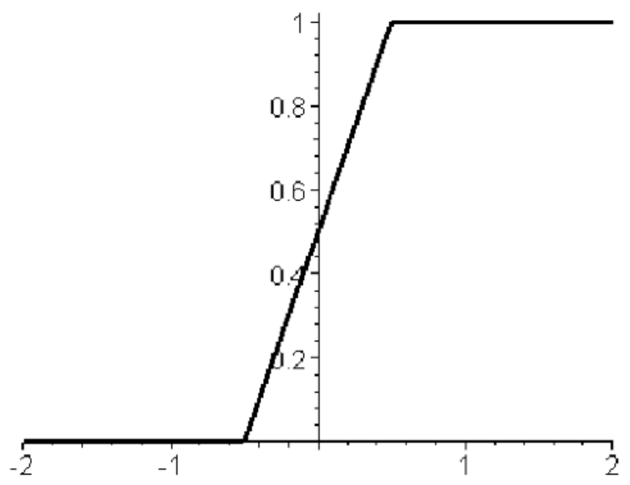
(1) *Threshold function*:

$$\varphi(x) = \begin{cases} 1 & x \geq 0 \\ 0 & x < 0 \end{cases}$$



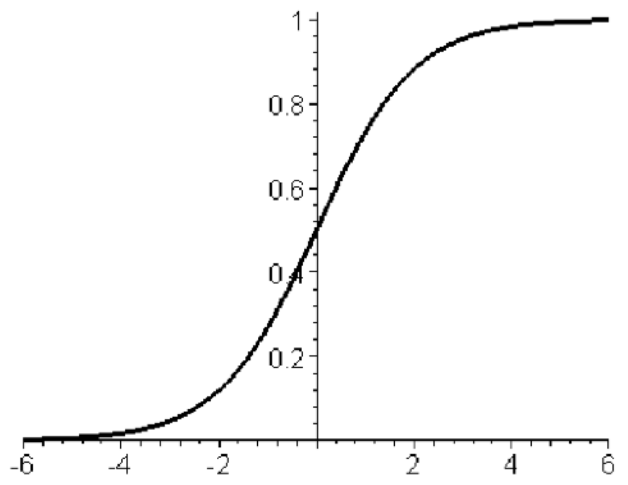
(2) *Piecewise linear function*:

$$\varphi(x) = \begin{cases} 1 & x \geq +\frac{1}{2} \\ x + \frac{1}{2} & \frac{1}{2} \geq x \geq -\frac{1}{2} \\ 0 & -\frac{1}{2} \geq x \end{cases}$$



(3) *Sigmoid function (logistic function)*:

$$\varphi(x) = \frac{1}{1 + e^{-x}}$$



Neural Net Training

Given is a continuous function

$$u : \Omega = [-1,1]^m \longrightarrow \mathbb{R}$$

that is sampled on a set $P \subseteq \Omega$.

Consider a neural net

$$\hat{u} : \Omega \times \Theta \longrightarrow \mathbb{R}\,, \quad \hat{u} = \hat{u}(x, W, B)$$

depending on x and weights & bias

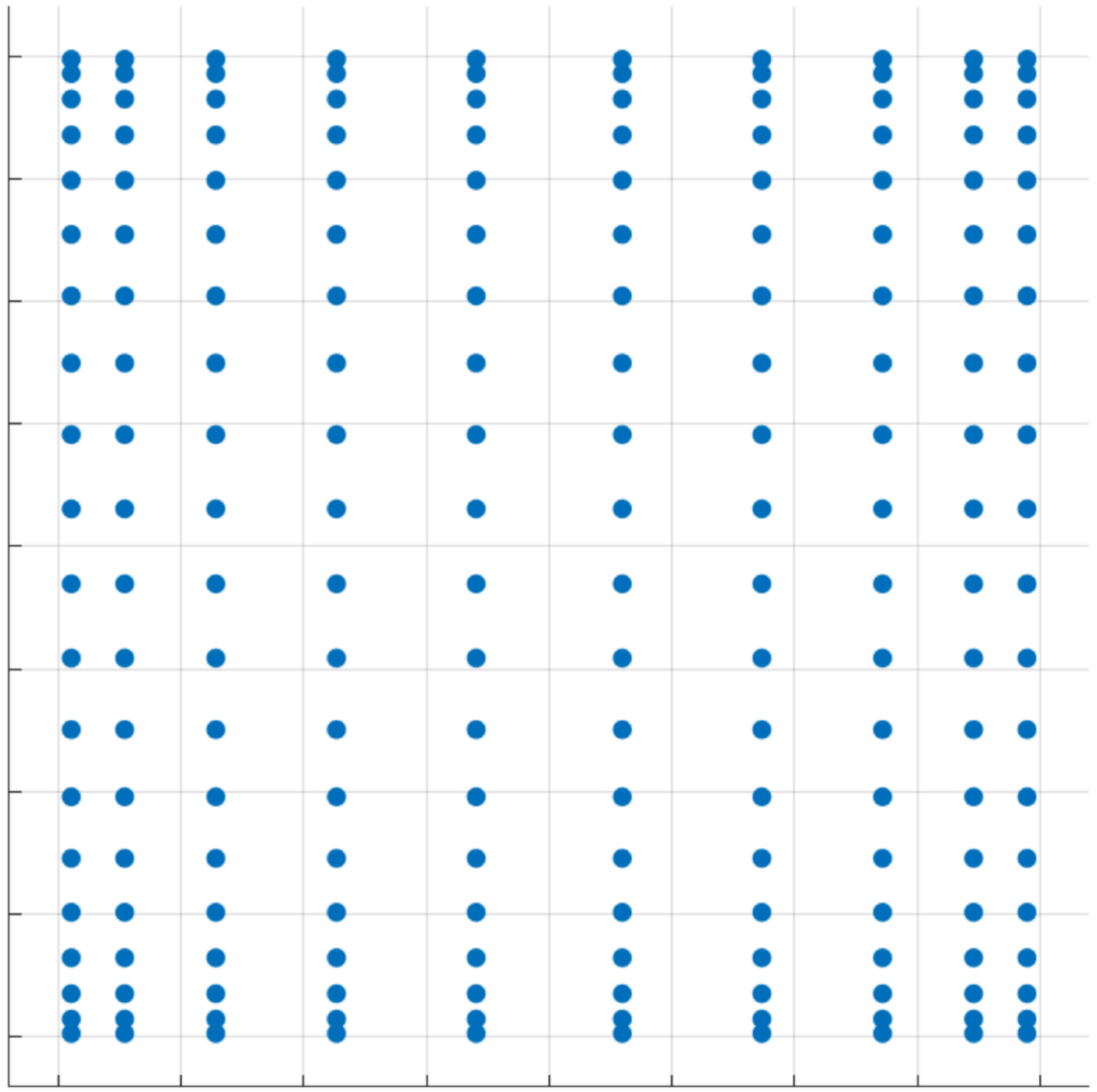
The mean square error (MSE) loss

$$\mathscr{L}(W,B) : \Theta \longrightarrow \mathbb{R}^+$$

is given by

$$\mathscr{L}(W,B) = \sum_{p \in P} \|u(p) - \hat{u}(p, W, B)\|^2 \approx \int_\Omega \|u(x) - \hat{u}(x, W, B)\|^2 d\Omega.$$

Loss approximations due to Gauss quadratures



Legendre grid in 2D

$$p_\alpha = (p_{\alpha_1}, \cdots, p_{\alpha_m}) \in G_{m,n}$$

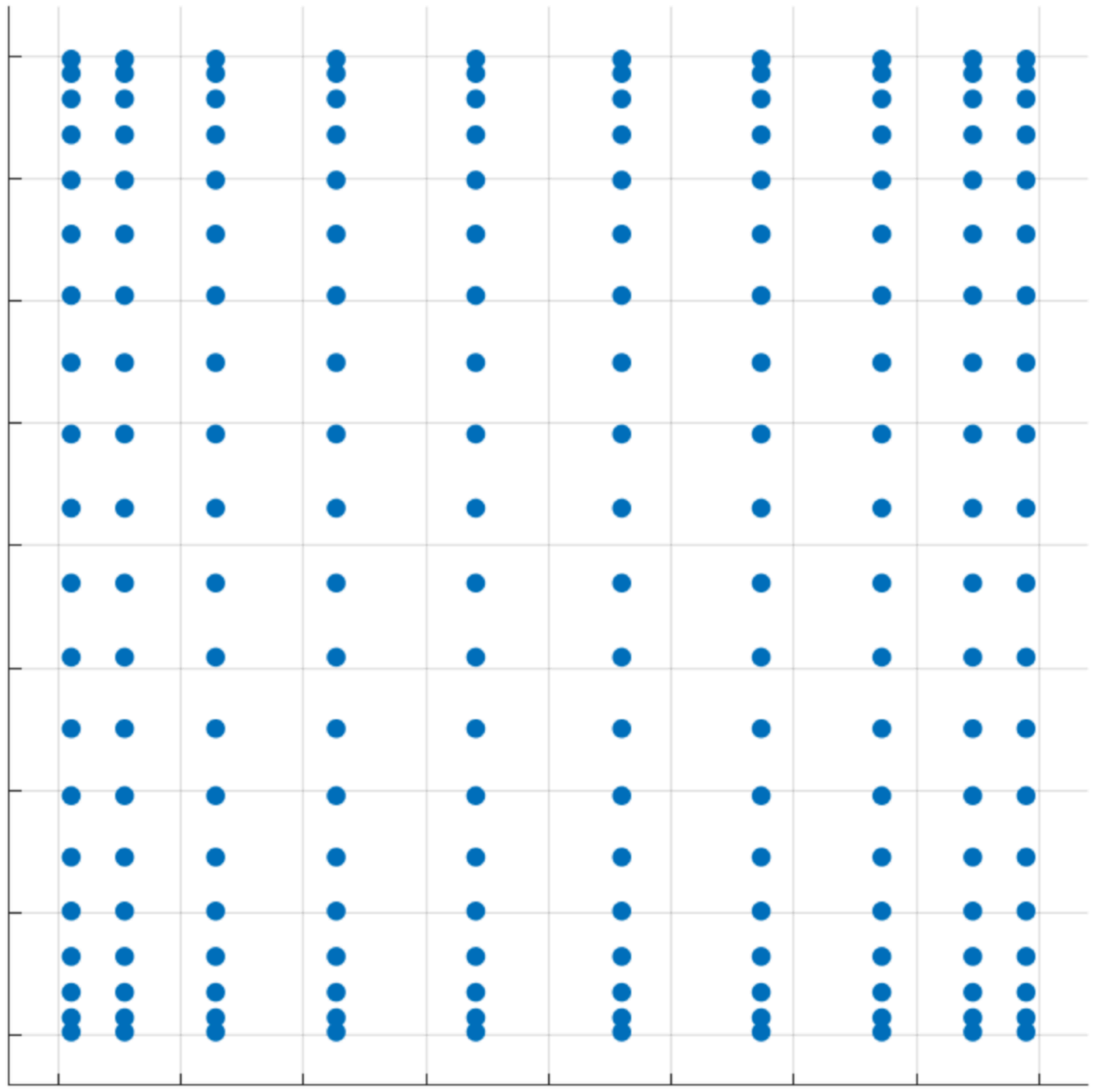
Lagrange polynomials in dimension $m \in \mathbb{N}$

$$A_{m,n} = \{\alpha \in \mathbb{N}^m : \|\alpha\|_1 \leq n\}$$

$$L_\alpha = \prod_{i=1}^{m} l_{\alpha_i,i}, \quad l_{j,i} = \prod_{h \neq j, h=0}^{n} \frac{x_i - p_h}{p_j - p_h}$$

$$L_\alpha(p_\beta) = \delta_{\alpha,\beta}$$

Loss approximations due to Gauss quadratures



Legendre grid in 2D

$$p_\alpha = (p_{\alpha_1}, \cdots, p_{\alpha_m}) \in G_{m,n}$$

Lagrange polynomials in dimension $m \in \mathbb{N}$

$$f(x) = \sum_{\alpha \in A_{m,n}} f(p_\alpha) L_\alpha = Q_{f,n}\,,$$

$$g(x) = \sum_{\alpha \in A_{m,n}} g(p_\alpha) L_\alpha = Q_{f,n}$$

$$\int_\Omega f(x)g(x)dx = \sum_{\alpha \in A_{m,n}} w_\alpha f(p_\alpha) g(p_\alpha)$$

w_α...Gauss-Legendre weights

Polynomial differentiation P.D.

Lagrange polynomials in dimension $m \in \mathbb{N}$

$$f(x) = \sum_{\alpha \in A_{m,n}} f(p_\alpha) L_\alpha = Q_{f,n}\,,$$

$$\partial^\alpha f(x) = \sum_{\alpha \in A_{m,n}} c_\alpha L_\alpha(x)\,, \quad C = (c_\alpha)_{\alpha \in A_{m,n}} = \mathbb{D}^\alpha F\,, F = (f(p_\alpha))_{\alpha \in A_{m,n}}$$

$\mathbb{D}^\alpha$...Polynomial Differential Operator

$$\mathbb{D}^\alpha = (\partial^\alpha L_\beta)_{\beta \in A_{m,n}} \in \mathbb{R}^{|A_{m,n}| \times |A_{m,n}|}\,, \quad \partial^\alpha L_\beta = \sum_{\gamma \in A_{m,n}} c_\gamma L_\gamma$$

Sobolev spaces

Sobolev spaces of regularity $k \in \mathbb{N}$

$$H^k(\Omega, \mathbb{R}) = \{f \in L^2(\Omega, \mathbb{R}) : \|\partial^\alpha f\|_{L^2(\Omega)} \leq \infty \, , \alpha \in \mathbb{N}^m \, , \|\alpha\|_1 \leq n\} \, ,$$

$$H^0(\Omega, \mathbb{R}) = L^2(\Omega, \mathbb{R})$$

$$\langle f, g \rangle_{H^k(\Omega)} = \sum_{\|\alpha\|_1 \leq n} \int_\Omega \partial^\alpha f(x) \partial^\alpha g(x) dx$$

Sobolev cubatures

Given polynomials $f(x) = \sum_{\alpha \in A_{m,n}} f(p_\alpha) L_\alpha = Q_{f,n}\,, g(x) = \sum_{\alpha \in A_{m,n}} g(p_\alpha) L_\alpha = Q_{f,n}$

$$\langle f, g \rangle_{H^k(\Omega)} = \sum_{\|\alpha\|_1 \leq n} (\mathbb{D}^\alpha F)^T W_{m,n} \mathbb{D}^\alpha G\,,$$

$$F = (f(p_\alpha))_{\alpha \in A_{m,n}}\,, G = (g(p_\alpha))_{\alpha \in A_{m,n}} \quad W_{m,n} = \mathrm{diag}(w_\alpha)_{\alpha \in A_{m,n}}$$

Loss Approximations

We are seeking for a smooth function solving a Poisson problem:

$\Delta u = 0$ **in** $\Omega = [-1,1]^m$

Poisson-PDE loss of neural nets with Automatic differentiation

$$\mathscr{L}(W,B) = \sum_{p \in P} \|\Delta \hat{u}(p,W,B)\|^2 \approx \int_\Omega \|\Delta \hat{u}(x,W,B)\|^2 d\Omega$$

Poisson-PDE loss of neural nets with Polynomial differentiation

$$\mathscr{L}(W,B) = (\mathbb{D}_\Delta \hat{U}(p_\alpha))^T W_{m,n} (\mathbb{D}_\Delta \hat{U}(p_\alpha)) \approx \|\Delta \hat{u}(x,W,B)\|^2_{L^2(\Omega)}.$$

$$\mathscr{L}(W,B) = \sum_{\|\beta\| \leq k} (\mathbb{D}^\beta \mathbb{D}_\Delta \hat{U}(p_\alpha))^T W_{m,n} \mathbb{D}_\Delta \mathbb{D}^\beta \hat{U}(p_\alpha)) \approx \|\Delta \hat{u}(x,W,B)\|^2_{H^k(\Omega)}$$

$\mathbb{D}_\Delta = \mathbb{D}^{2,0,\ldots,0} + \cdots + \mathbb{D}^{0,\cdots,0,2}$ **approximates the Laplacian**

Approximation Errors

Given a regular k-times differentiable function

$f : \Omega \longrightarrow \mathbb{R}$

Mean square error

$$\sum_{p \in P} \|f(p)\|^2 - \int_\Omega \|f(x)\|^2 d\Omega \in \mathcal{O}\left(\frac{1}{\sqrt{|P|}}\right)$$

Gauss cubature error

$$F^T W_{m,n} F - \int_\Omega \|f(x)\|^2 d\Omega \in \mathcal{O}\left(\frac{1}{k(n-k)^k}\right), \quad |A_{m,n}| = (n+1)^m$$

In low dimensions Gauss quadrature is more powerful than Monte Carlo integration !

Runtime Complexity

Theorem 4. *For a given deep Neural Network $\hat{u}_\theta : \Omega \to \mathbb{R}$, with architecture $\xi_{m,1}$ consisting of l hidden layers and q neurons per layer, the complexity per epoch for computing the $k-$th derivative $(\partial_x^k \hat{u}_\theta)$ in $s \in \mathbb{N}$ training points is given by*

i) $\mathcal{O}(2^{k-1} lsq^2)$ *for a PINN resting on A.D., i.e. it scales exponentially with the order of the derivative.*

ii) $\mathcal{O}(\max\{s^2, lsq^2\})$ *for the SC-PINN using P.D.*

In low dimensions $m \leq 4$ Polynomial Differentiation (P.D.) is faster than Automatic Differentiation (A.D.) !

Cardona, J. E. S., & Hecht, M. (2022). Replacing Automatic Differentiation by Sobolev Cubatures fastens Physics Informed Neural Nets and strengthens their Approximation Power. *arXiv preprint arXiv:2211.15443*.

Examples

We start by addressing the Poisson problem in dimension $m = 2$

$$\begin{cases} -\Delta u(x) - f(x) &= 0 \quad , \forall x \in \Omega = [-1,1]^2 \\ u(x) - g(x) &= 0 \quad , \forall x \in \partial\Omega \,, \end{cases} \tag{30}$$

described in detail Eq. (11). We choose $f(x,y) := -2\lambda^2 \cos(\lambda x)\sin(\lambda y)$ and $g(x) := \cos(\lambda x)\sin(\lambda y)$, with frequency $\lambda = 2\pi q$, $q = 6$, yielding $u(x,y) = g(x,y)$ to be the analytic solution.

Cardona, J. E. S., & Hecht, M. (2022). Replacing Automatic Differentiation by Sobolev Cubatures fastens Physics Informed Neural Nets and strengthens their Approximation Power. *arXiv preprint arXiv:2211.15443*.

Examples

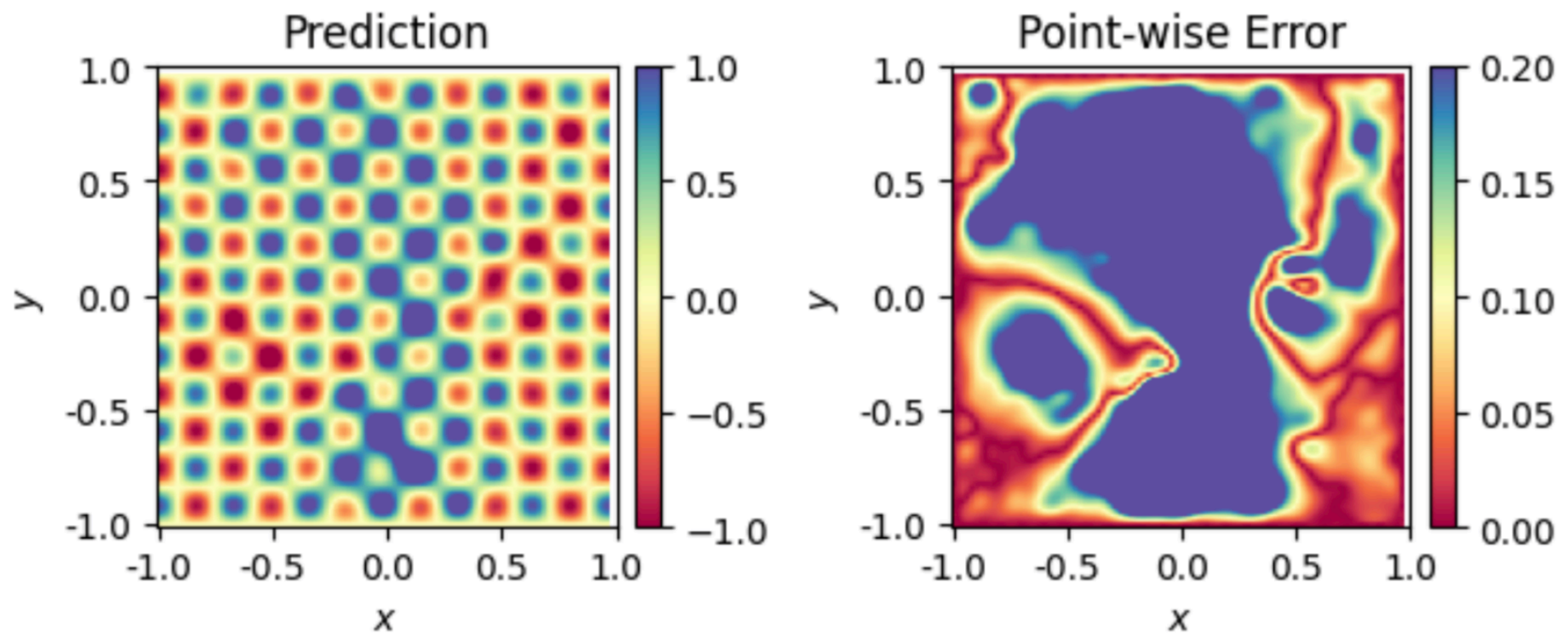


Figure 1. ID-PINN with MSE loss, reaching $\epsilon_\infty = 1.24$, $\epsilon_1 = 2.22\text{e-}1$, $t \approx 1173$.

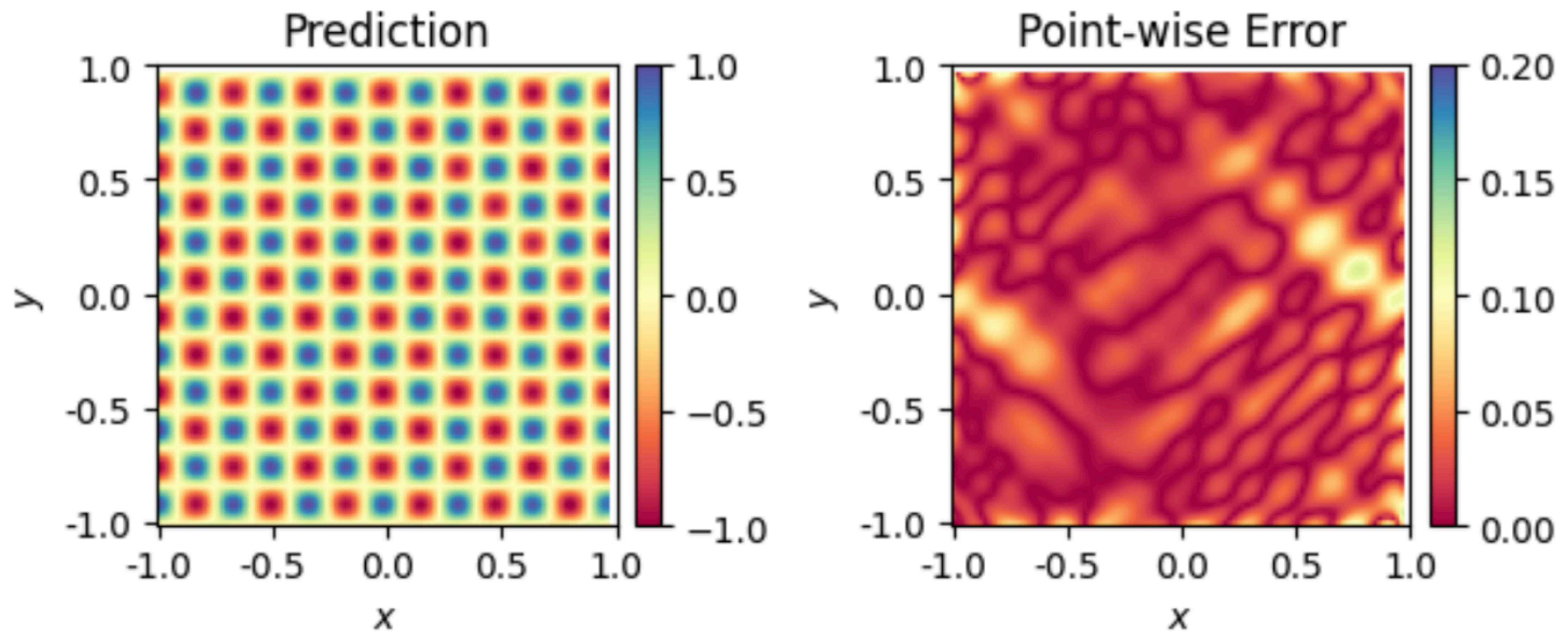


Figure 2. SC-PINN with strong Sobolev loss and A.D., reaching $\epsilon_\infty = 1.37\text{e-}1$, $\epsilon_1 = 2.27\text{e-}2$, $t \approx 725$.

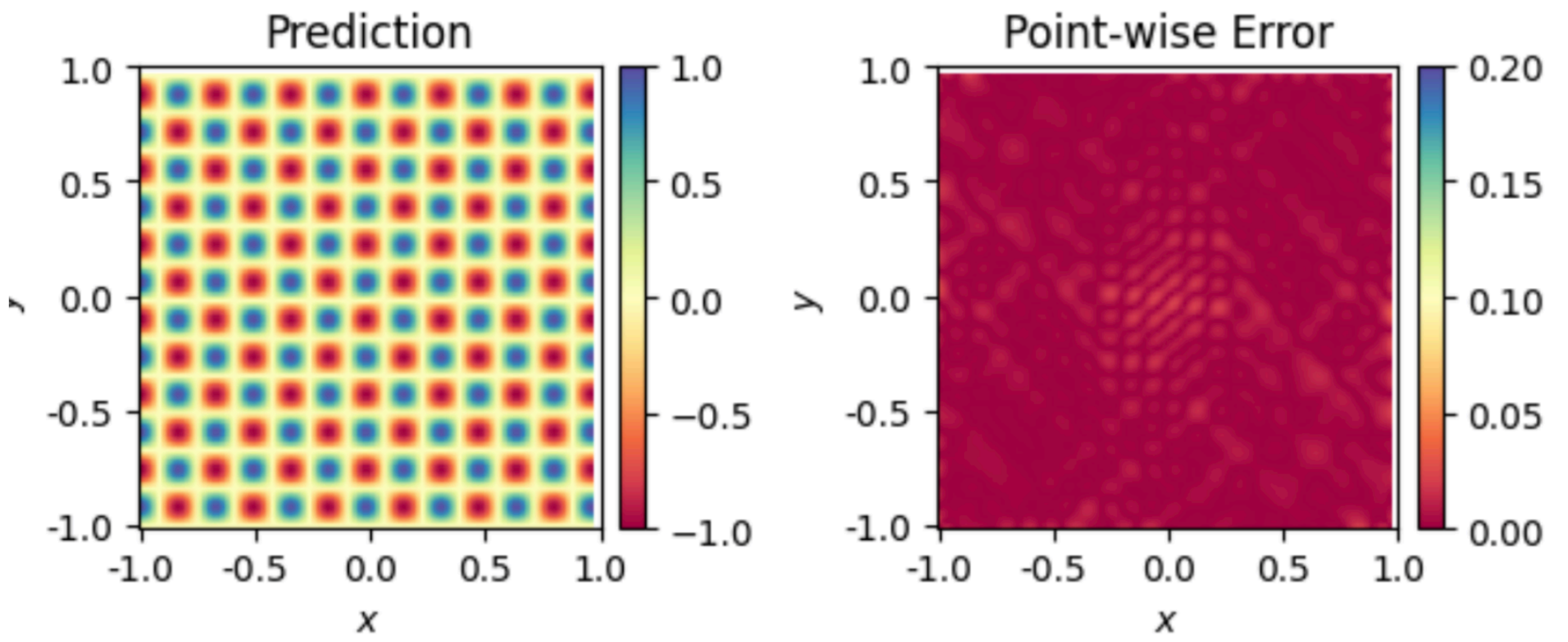


Figure 3. SC-PINN with strong Sobolev loss and P.D., reaching $\epsilon_\infty = 3.00\text{e-}2$, $\epsilon_1 = 3.75\text{e-}3$, $t \approx 192$.

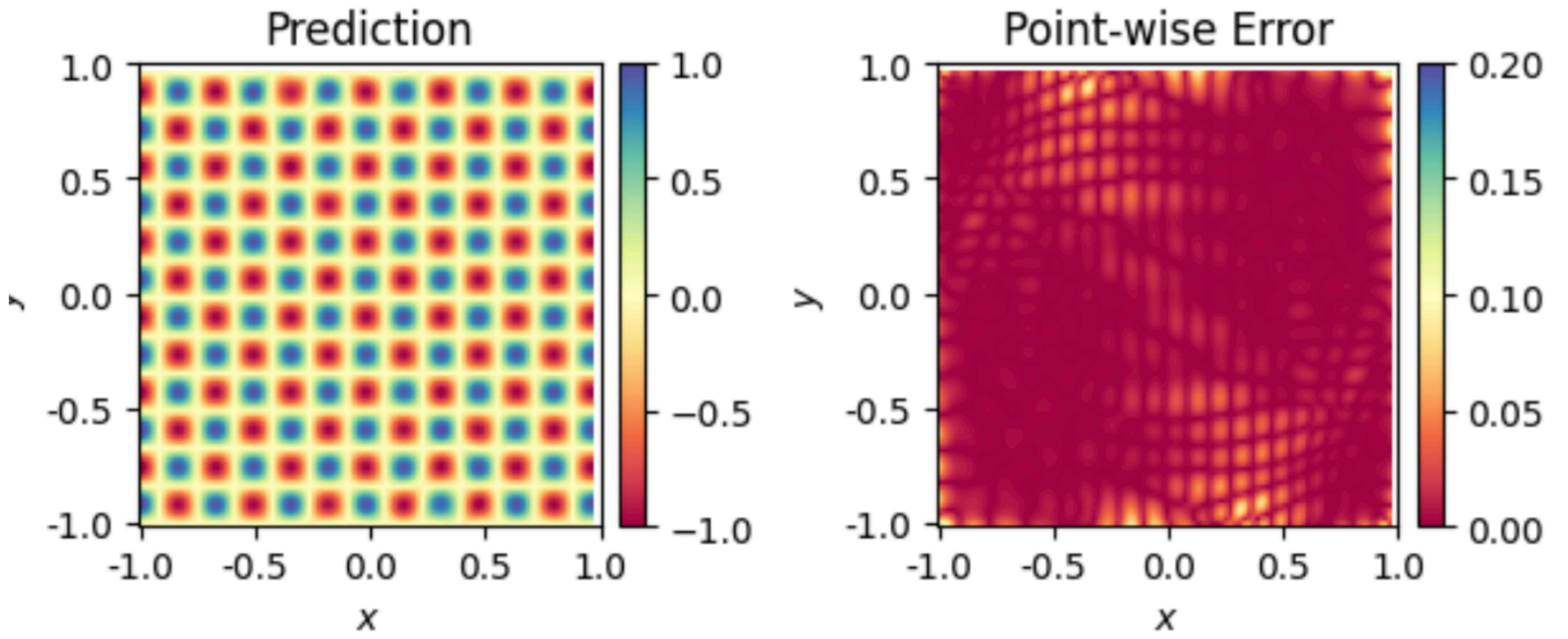


Figure 4. SC-PINN with strong variational Sobolev loss and P.D., reaching $\epsilon_\infty = 1.25\text{e-}1$, $\epsilon_1 = 7.70\text{e-}3$, $t \approx 188$.

Cardona, J. E. S., & Hecht, M. (2022). Replacing Automatic Differentiation by Sobolev Cubatures fastens Physics Informed Neural Nets and strengthens their Approximation Power. *arXiv preprint arXiv:2211.15443*.

Examples

The time independent *Quantum Harmonic Oscillator* in dimension $m = 2$, corresponds to the *Schrödinger equation* with the linear potential $V(u(x)) := (x_1^2 + x_2^2)u(x)$, $u \in C^2(\Omega, \mathbb{R})$, see e.g. (Griffiths and Schroeter, 2018; Liboff, 1980), given by:

$$\begin{cases} -\Delta u(x) + V(u(x)) &= \lambda u(x) \quad ,\forall x \in \Omega \\ u(x) - g(x) &= 0 \quad ,\forall x \in \partial\Omega\,, \end{cases} \tag{31}$$

We consider the eigenvalue problem $\lambda = n_1 + n_2 + 1$, $n_1, n_2 \in \mathbb{N}$ with eigenfunctions

$$g(x_1, x_2) = \frac{\pi^{-1/4}}{\sqrt{2^{n_1+n_2} n_1! n_2!}} e^{-\frac{(x_1^2+x_2^2)}{2}} H_{n_1}(x_1) H_{n_2}(x_2)\,,$$

We choose $\lambda = 15$

Cardona, J. E. S., & Hecht, M. (2022). Replacing Automatic Differentiation by Sobolev Cubatures fastens Physics Informed Neural Nets and strengthens their Approximation Power. *arXiv preprint arXiv:2211.15443*.

Examples

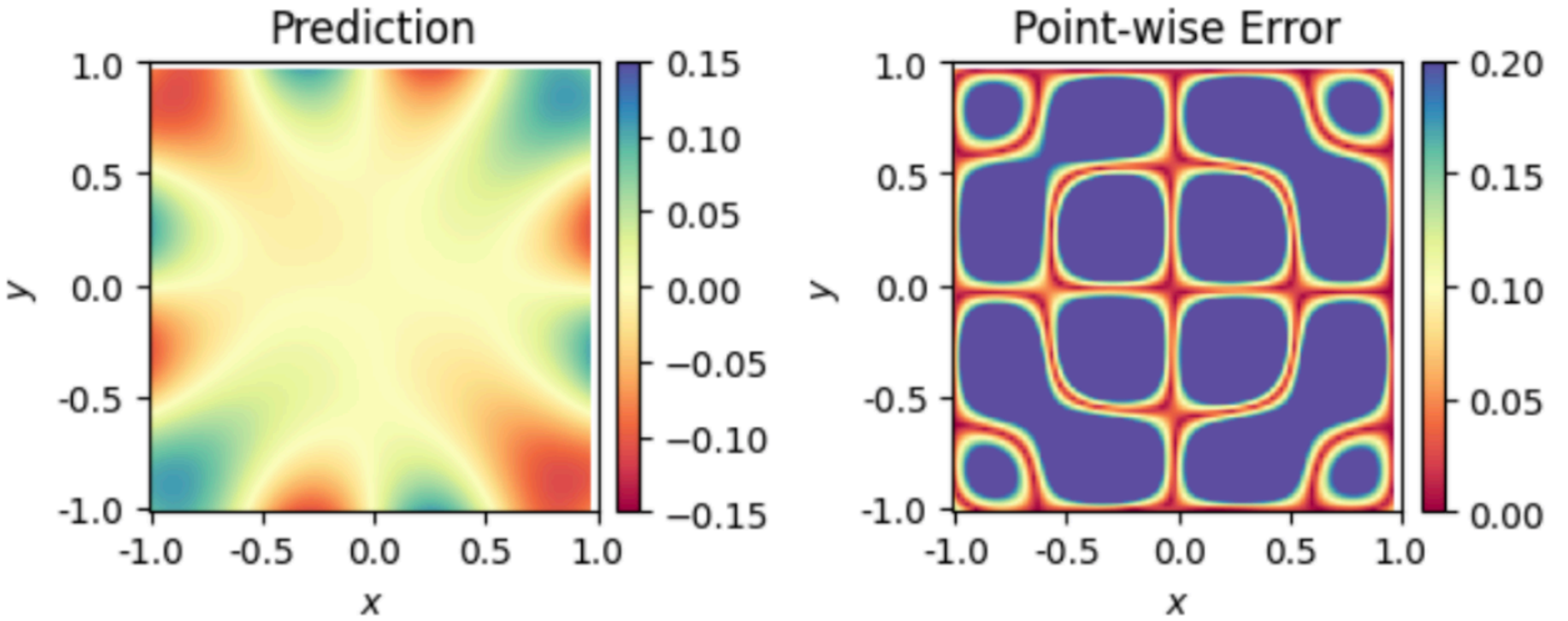


Figure 5. ID-PINN with MSE loss and A.D., reaching $\epsilon_\infty = 1.46\text{e-}1$, $\epsilon_1 = 4.78\text{e-}2$, $t \approx 905$,

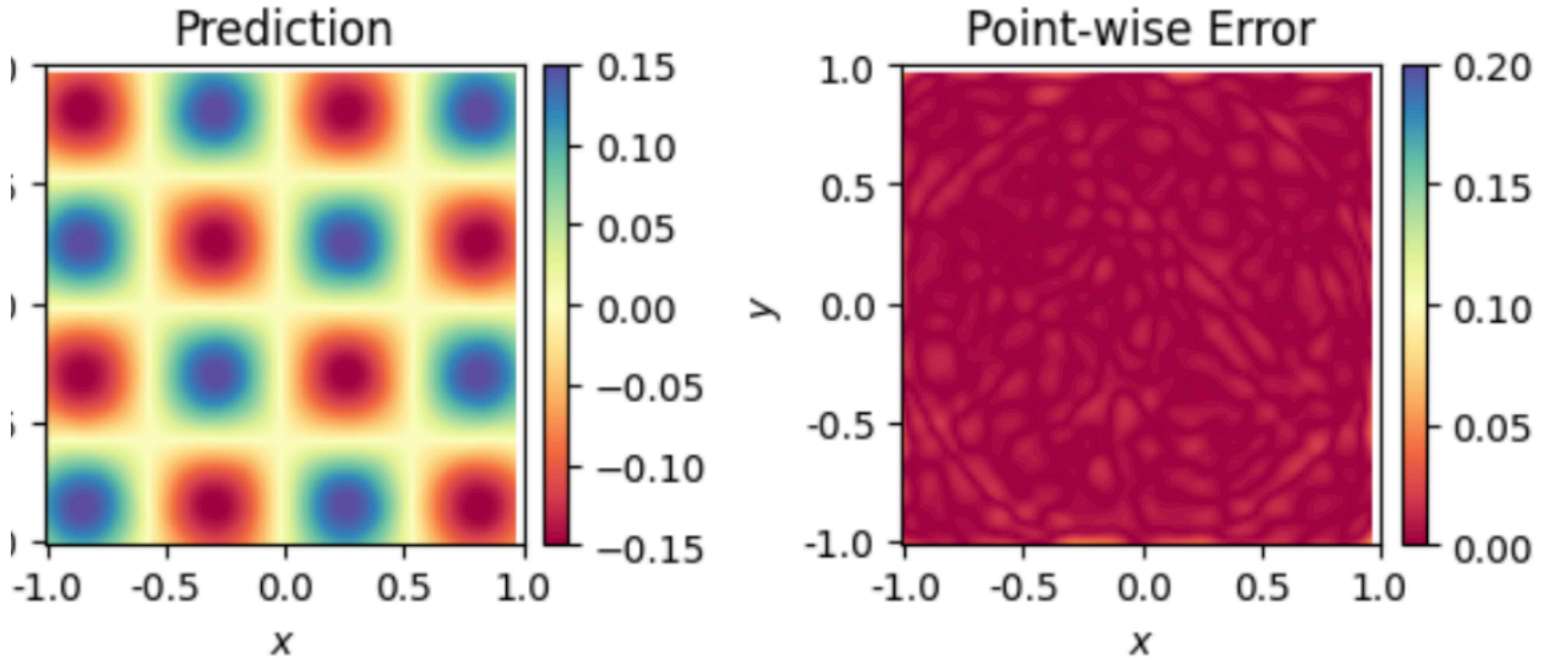


Figure 7. SC-PINN with strong variational Sobolev loss and P.D., reaching, $\epsilon_\infty = 7.27\text{e-}3, \epsilon_1 = 8.16\text{e-}4, t \approx 167$,

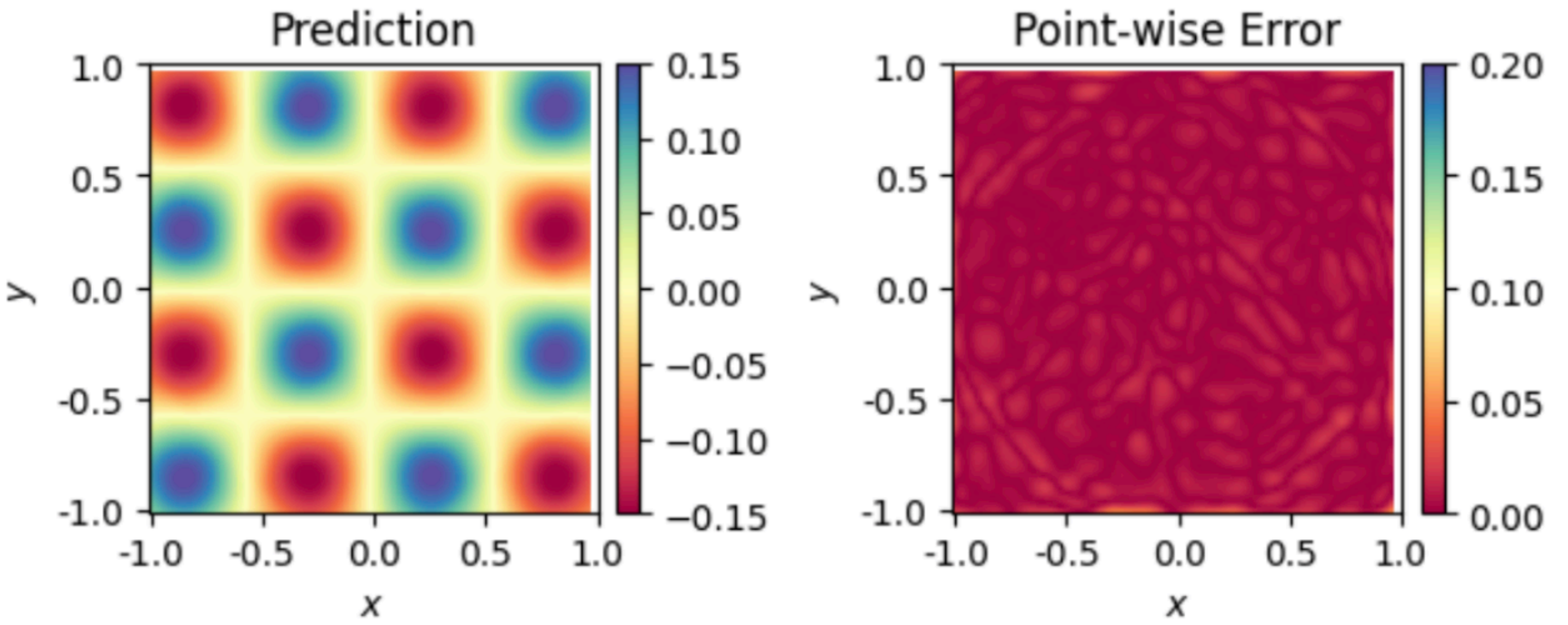


Figure 6. SC-PINN with strong Sobolev loss and P.D., reaching, $\epsilon_\infty = 1.22\text{e-}2$, $\epsilon_1 = 1.24\text{e-}3$, $t \approx 165$,

Cardona, J. E. S., & Hecht, M. (2022). Replacing Automatic Differentiation by Sobolev Cubatures fastens Physics Informed Neural Nets and strengthens their Approximation Power. *arXiv preprint arXiv:2211.15443*.

Examples

We re-investigate PINN-solutions of the Poisson problem in dimension $m = 1$, whose analytic solutions include hard transitions. That is, choosing

$$f(x) := C\left(A\omega^2 \sin(\omega x) + 2\beta^2 \text{sech}^2(x) \tanh(\beta x)\right),$$

with boundary condition $g(x) := C(A \sin(\omega x) + \tanh(\beta x))$ yielding $u(x) = g(x)$ to be the analytic solution. Two scenarios were considered:

$$S_1 = \{C = 0.1, A = 0.1, \beta = 30, \omega = 20\pi, \text{bs} = 100\}$$
$$S_2 = \{C = 0.1, A = 0.1, \beta = 5, \omega = 26.5\pi, \text{bs} = 100\}$$

Cardona, J. E. S., & Hecht, M. (2022). Replacing Automatic Differentiation by Sobolev Cubatures fastens Physics Informed Neural Nets and strengthens their Approximation Power. *arXiv preprint arXiv:2211.15443*.

Examples

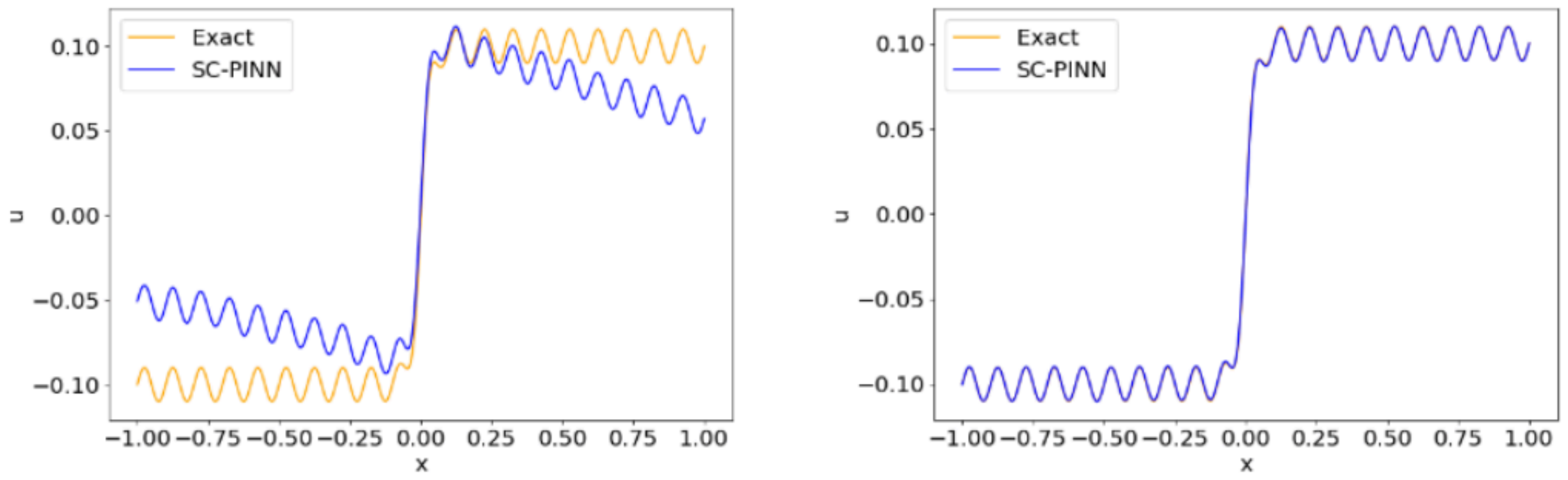


Figure 8. SC-PINN with strong Sobolev loss and P.D. (left), reaching $\epsilon_\infty = 3.04e-2$, $\epsilon_1 = 7.24e-2, t \approx 150$, SC-PINN with strong variational Sobolev loss and P.D.(right), reaching $\epsilon_\infty = 2.0\text{e-}3$, $\epsilon_1 = 4.0\text{e-}4, t \approx 151$, scenario S_1.

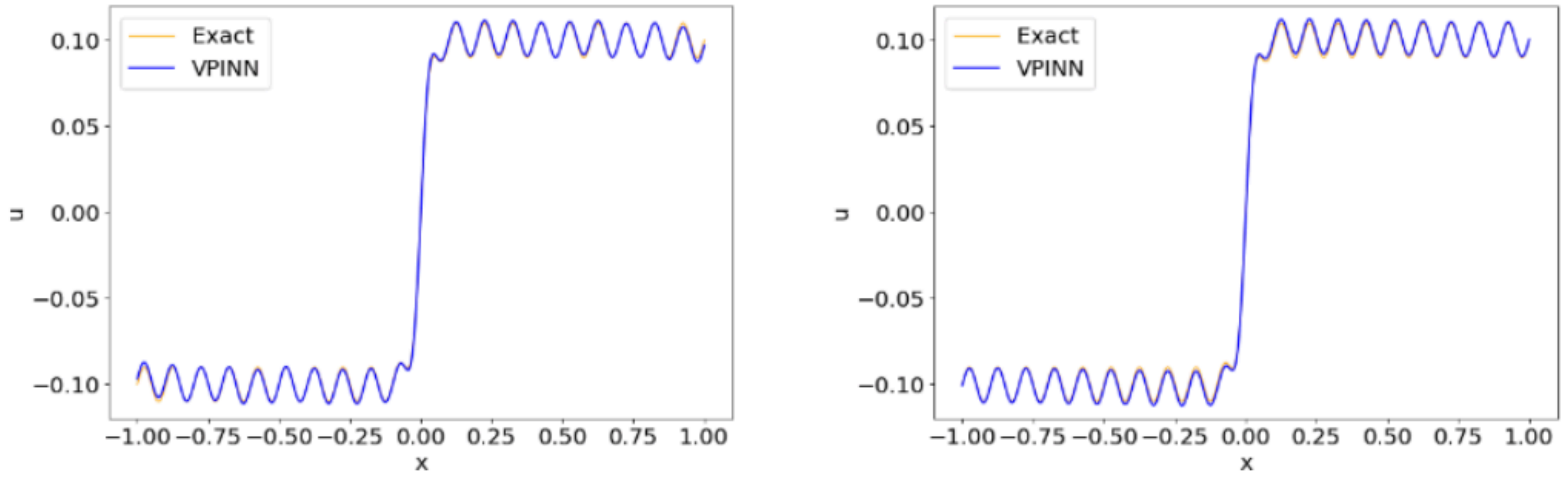


Figure 9. VPINN with $N_{\text{el}} = 1$ (left), reaching $\epsilon_\infty = 3.29\text{e-}3$, $\epsilon_1 = 9.94\text{e-}4, t \approx 96$, VPINN with $N_{\text{el}} = 3$ (right), reaching $\epsilon_\infty = 2.73\text{e-}3$, $\epsilon_1 = 1.40\text{e-}3, t \approx 191$, scenario S_1.

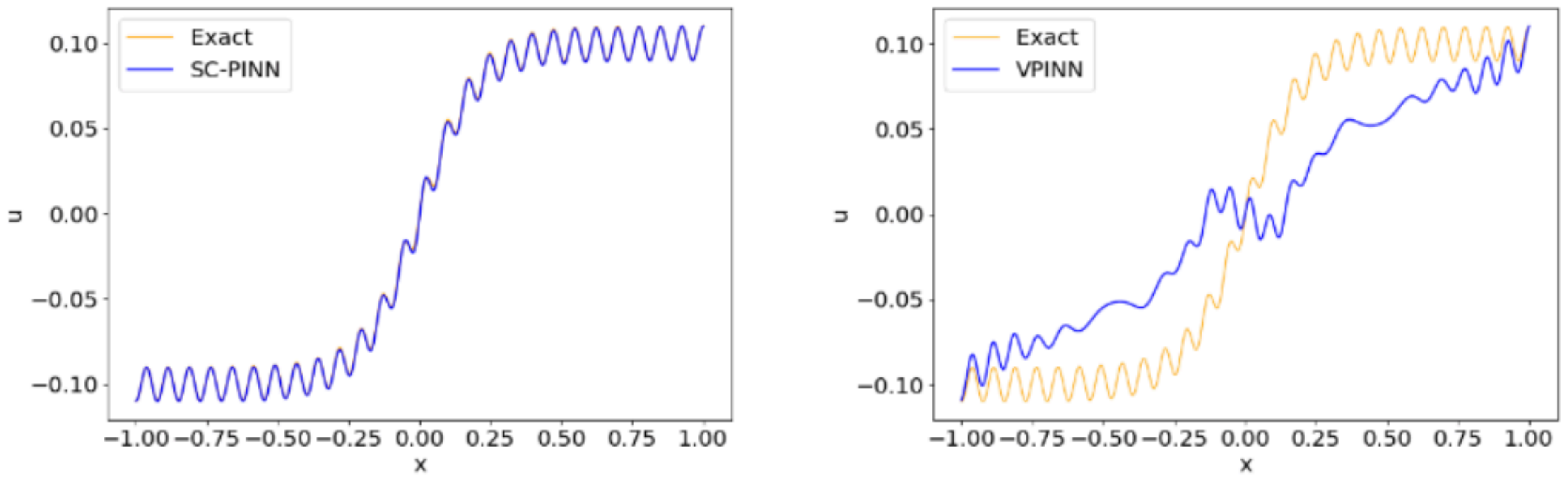


Figure 10. SC-PINN with strong variational Sobolev loss and P.D.(left), reaching $\epsilon_\infty = 2.75\text{e-}3$, $\epsilon_1 = 5.35\text{e-}4, t \approx 151$, VPINN with $N_{\text{el}} = 3$, reaching $\epsilon_\infty = 6.50\text{e-}2$, $\epsilon_1 = 3.40\text{e-}2$, scenario S_2, $t \approx 180$.

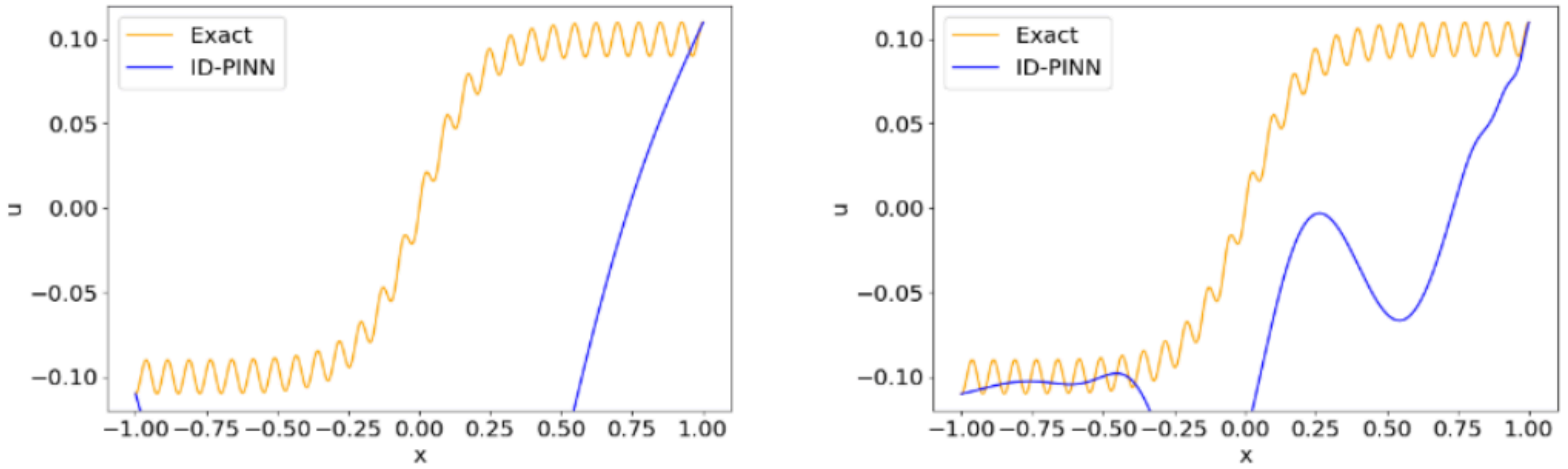
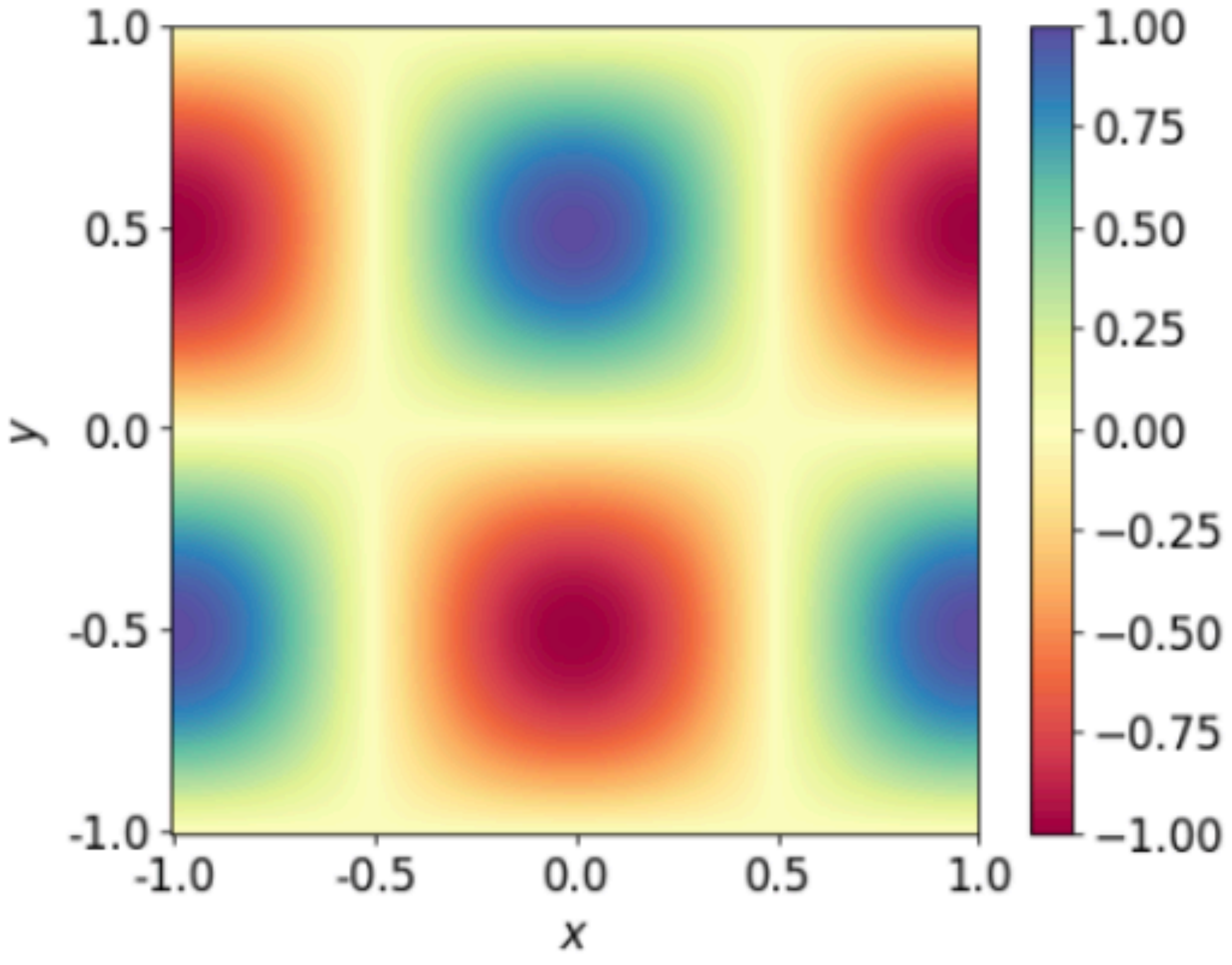


Figure 11. ID-PINN with MSE loss, bs = 500 and A.D.,(left), reaching $\epsilon_\infty = 4.74\text{e-}1$, $\epsilon_1 = 2.50\text{e-}1, t \approx 158$, ID-PINN with MSE loss without I.D., bs = 5000 and A.D.,(right), reaching $\epsilon_\infty = 1.73\text{e-}1$, $\epsilon_1 = 7.13\text{e-}2, t \approx 360$, scenario S_2.

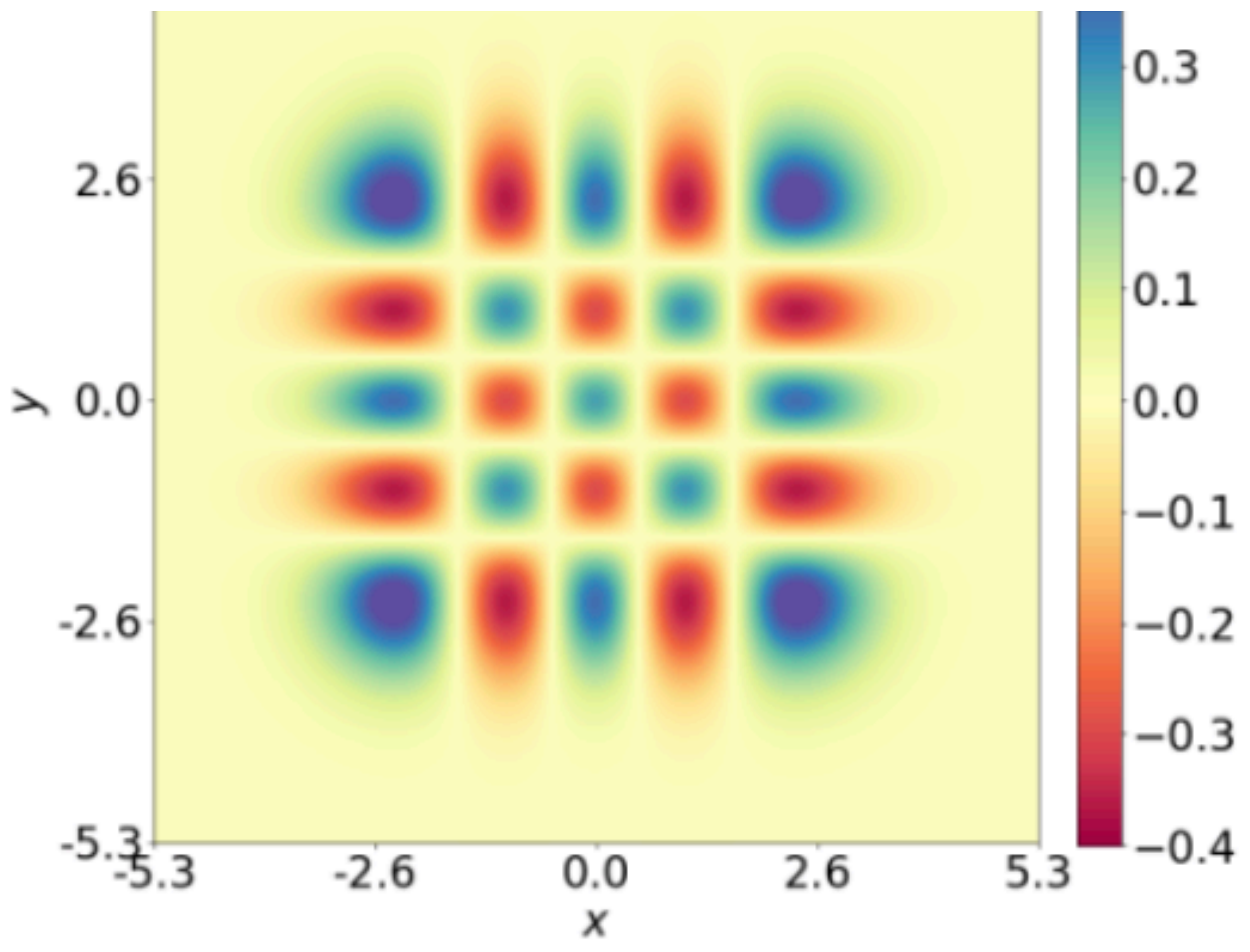
Cardona, J. E. S., & Hecht, M. (2022). Replacing Automatic Differentiation by Sobolev Cubatures fastens Physics Informed Neural Nets and strengthens their Approximation Power. *arXiv preprint arXiv:2211.15443*.

Inverse Eigenvalue Problems



Method	Approximation error		Runtime (s)
	ϵ_λ	ϵ_1	
PINN	$4.63 \cdot 10^{-1}$	$1.13 \cdot 10^{-2}$	$t \approx 1592$
ID-PINN	$2.14 \cdot 10^{-2}$	$8.09 \cdot 10^{-4}$	$t \approx 2184$
SC-PINN	$\mathbf{3 \cdot 10^{-4}}$	$\mathbf{5.49 \cdot 10^{-4}}$	$t \approx \mathbf{103}$

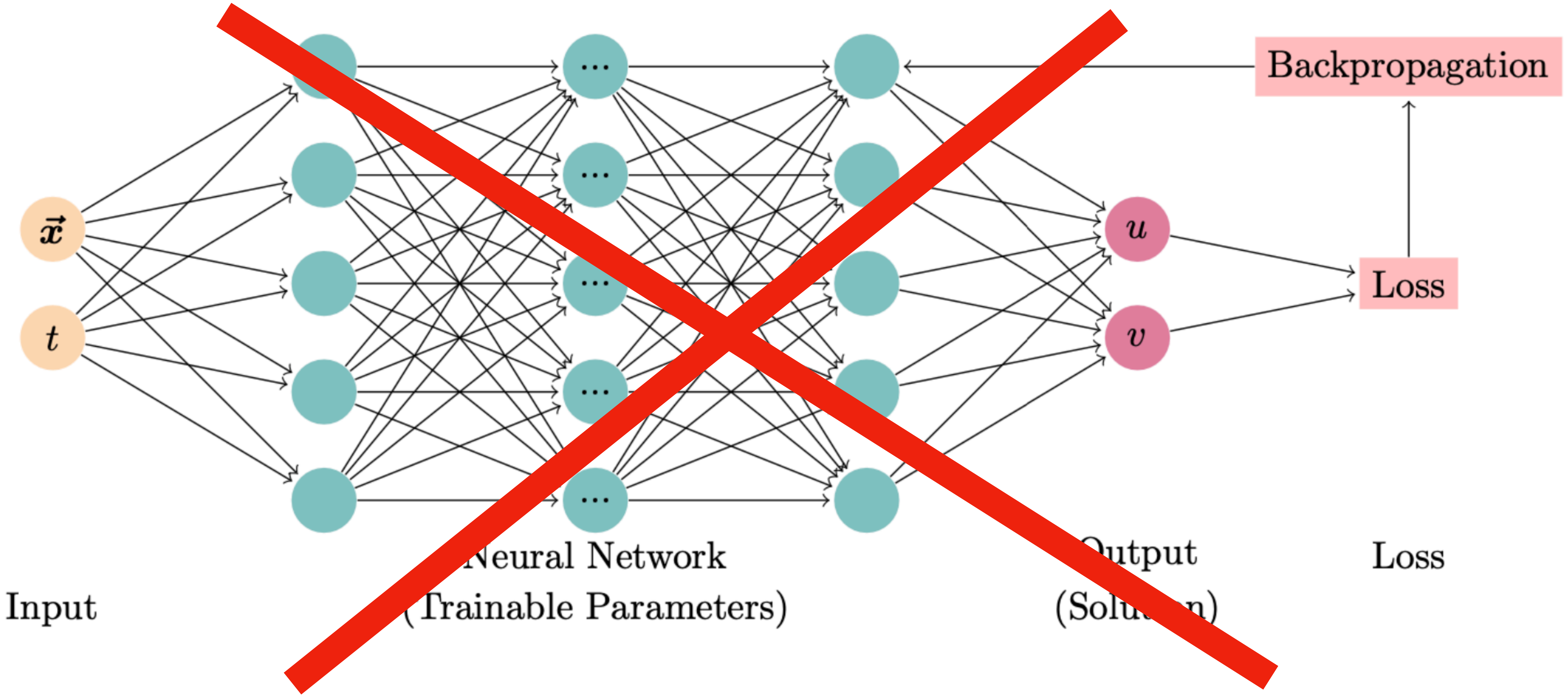
Figure 12. Solution and errors for 2D Poisson inverse problem with $\lambda_{gt} = \pi$.



Method	Approximation error		Runtime (s)
	ϵ_λ	ϵ_1	
PINN	6.01	$7.32 \cdot 10^{-2}$	$t \approx 1414$
ID-PINN	$6.21 \cdot 10^{-2}$	$7.51 \cdot 10^{-3}$	$t \approx 1346$
SC-PINN	$\mathbf{2.18 \cdot 10^{-4}}$	$\mathbf{5.68 \cdot 10^{-4}}$	$t \approx \mathbf{192}$

Figure 13. Solution and errors for 2D QHO inverse problem with $\lambda_{gt} = 5$.

Replace the Neural Net by (Spectral) Polynomial Surrogate Models (e.g. Chebyshev Polynomials)



$$\hat{u}(x,\Theta) = \sum_{\alpha \in A_{m,n}} \theta_\alpha T_\alpha(x)\,, \quad \Theta = (\theta_\alpha)_{\alpha \in A_{m,n}} \in \mathbb{R}^{|A_{m,n}|}$$

$$T_\alpha(\cos(x)) = T_\alpha(\cos(x_1),\ldots,\cos(x_m)) = \prod_{i=1}^{m} \cos(\alpha_i x_i) = \cos(\alpha x)$$

(Spectral) Polynomial Surrogates Models (PSMs)
(Work in Progress)

$$\hat{u}(x,\Theta) = \sum_{\alpha \in A_{m,n}} \theta_\alpha T_\alpha(x)\,, \quad \Theta = (\theta_\alpha)_{\alpha \in A_{m,n}} \in \mathbb{R}^{|A_{m,n}|}$$

$$T_\alpha(\cos(x)) = T_\alpha(\cos(x_1),\dots,\cos(x_m)) = \prod_{i=1}^{m} \cos(\alpha_i x_i) = \cos(\alpha x)$$

Formulate losses with respect to Sobolev spaces of negative order

$$H^{-k}(\Omega,\mathbb{R}) := \overline{\mathscr{D}}^{\|\cdot\|_{H^{-k}(\Omega)}}\,, \quad \|F\|_{H^{-k}(\Omega)} = \sup_{u \in H^k(\Omega,\mathbb{R})} \frac{|Fu|}{\|u\|_{H^k(\Omega)}}$$

(Spectral) Polynomial Surrogates Models (PSMs)
(Work in Progress)

Experiment 4.1 (Non-periodic 2D-Poisson forward problem with hard transitions). *We consider the Poisson equation with right hand side function f given by*

$$
\begin{aligned}
f(x,y) =& C(A\sin(\omega y)+\tanh(\beta y))(-A\omega^2\sin(\omega x)-2\beta^2\tanh(\beta x)\mathrm{sech}^2(\beta x)) \\
&+C(A\sin(\omega x)+\tanh(\beta x))(-A\omega^2\sin(\omega y)-2\beta^2\tanh(\beta y)\mathrm{sech}^2(\beta y)),
\end{aligned}
$$

with $C=0.1, A=0.1, \beta=5, \omega=10\pi$. All the experiments where conducted with the same number of training points, as required for the Sobolev cubatures of degree $n=50$ in the domain and $n=100$ for the boundary. For the SC-PINN the weak L^2 -loss was used for the PDE loss and for the boundary one.

(Spectral) Polynomial Surrogates Models (PSMs)
(Work in Progress)

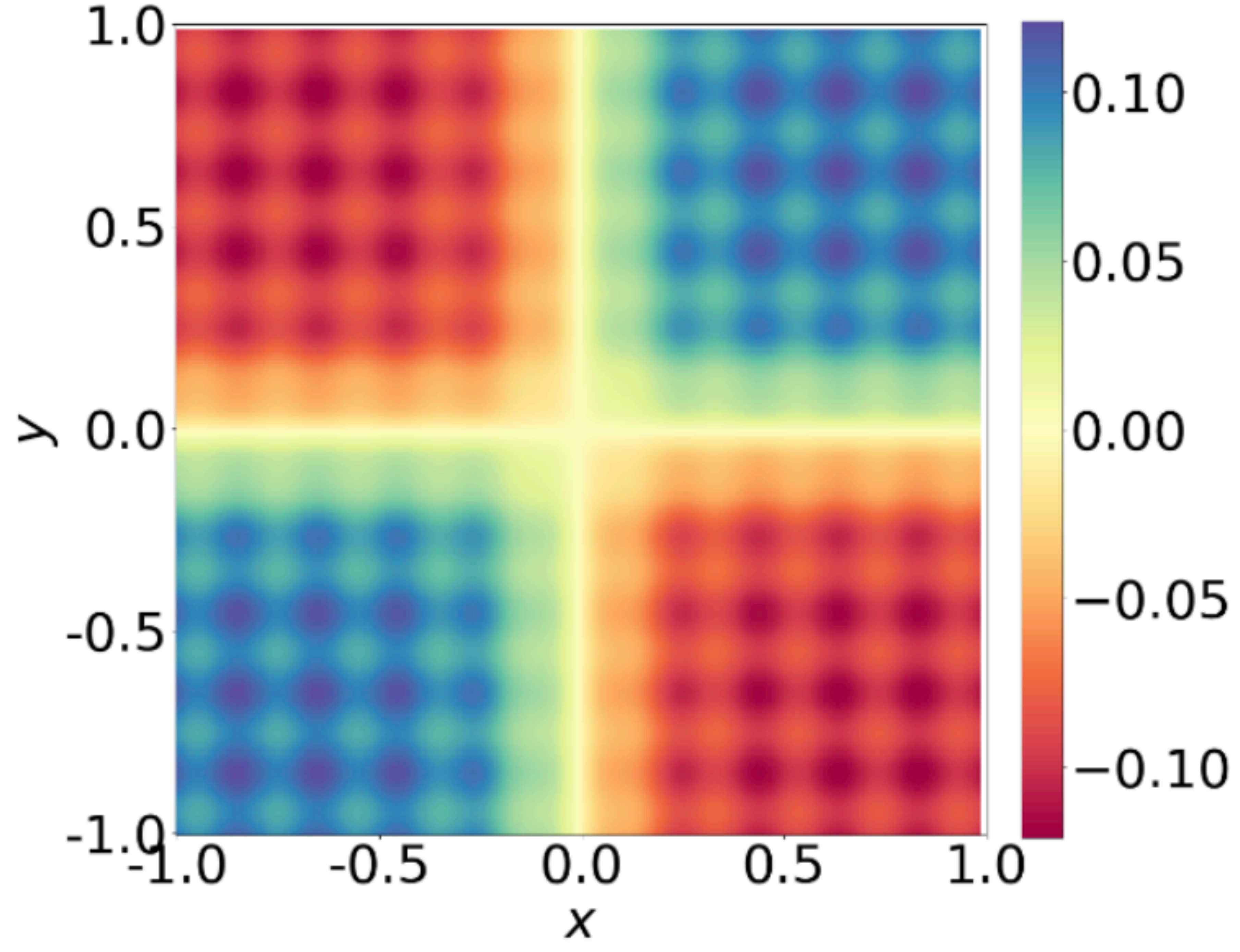


Figure 1: Solution for 2D Poisson problem

	Approximation error		Runtime (s)
dim = 2	ϵ_1	ϵ_∞	
PINN	$4.43 \cdot 10^{-3}$	$5.2 \cdot 10^{-2}$	$t = 886$
ID-PINN	$5.23 \cdot 10^{-3}$	$1.9 \cdot 10^{-2}$	$t = 1356$
SC-PINN	$2.52 \cdot 10^{-3}$	$3.33 \cdot 10^{-2}$	$t = 79.2$
GF-PSM	$5.37 \cdot 10^{-5}$	$2.94 \cdot 10^{-3}$	$t = 12.84$
AD-PSM	$\mathbf{8.79 \cdot 10^{-10}}$	$\mathbf{1.25 \cdot 10^{-8}}$	$\mathbf{t = 1.21}$

	Approximation error		Runtime (s)
dim = 4	ϵ_1	ϵ_∞	
GF-PSM	$1.33 \cdot 10^{-6}$	$1.0 \cdot 10^{-3}$	$t = 173.59s$
AD-PSM	$\mathbf{5.42 \cdot 10^{-8}}$	$\mathbf{6.37 \cdot 10^{-7}}$	$t = \mathbf{7.66s}$

Table 1: Errors for 2D and 4D Poisson forward problem

Poisson Inverse Eigenvalue Problem

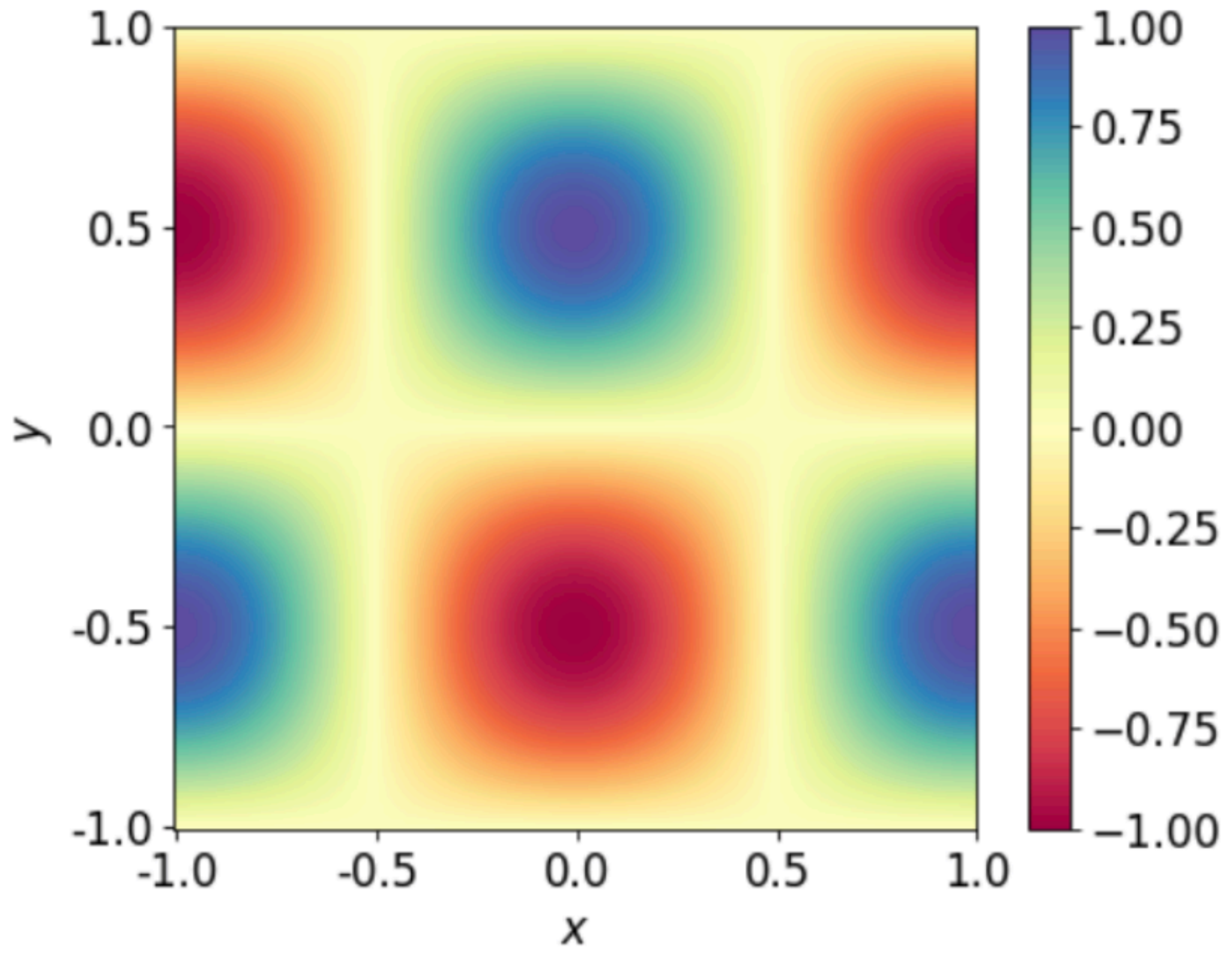


Figure 2: Solution for 2D inverse Poisson problem with $\omega_{gt} = \pi$.

	Approximation error		Runtime (s)
	ϵ_λ	ϵ_1	
PINN	$4.63 \cdot 10^{-1}$	$1.13 \cdot 10^{-2}$	$t \approx 1592$
ID-PINN	$2.14 \cdot 10^{-2}$	$8.09 \cdot 10^{-4}$	$t \approx 2184$
SC-PINN	$3 \cdot 10^{-4}$	$5.49 \cdot 10^{-4}$	$t \approx 103$
GF-PSM	$\mathbf{5.8 \cdot 10^{-8}}$	$\mathbf{6.0 \cdot 10^{-10}}$	$t \approx \mathbf{0.49}$

Table 2: Errors for 2D Poisson inverse problem

Quantum Harmonic Oscillator (QHO) forward Problem

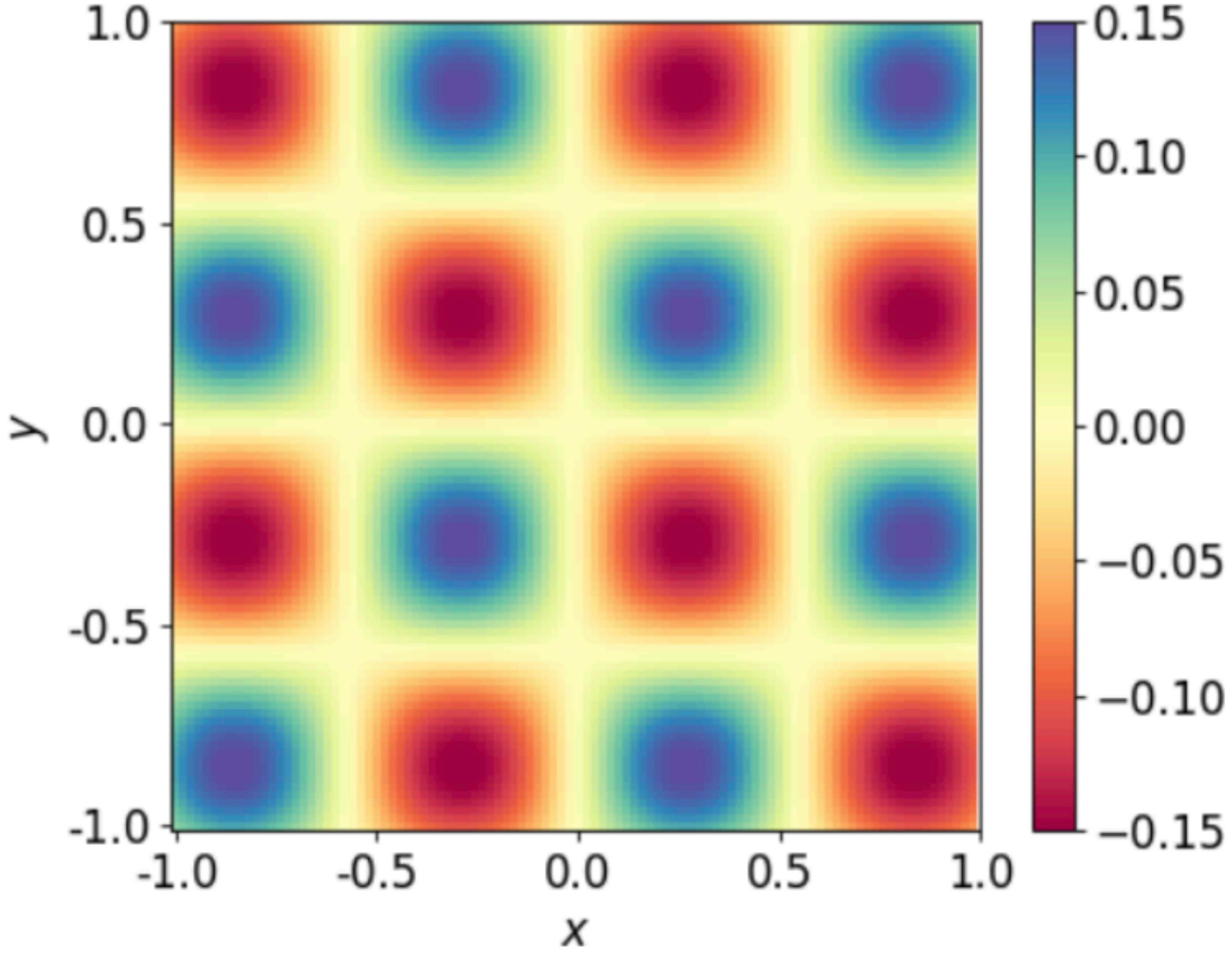


Figure 3: Solution of 2D QHO forward problem with $\mu = 21$.

	Approximation error		Runtime (s)
$\mu = 21$	ϵ_1	ϵ_∞	
PINN	$6.97 \cdot 10^{-2}$	$1. \cdot 10^{-3}$	$t \approx 776$
ID-PINN	$4.29 \cdot 10^{-2}$	$1.30 \cdot 10^{-1}$	$t \approx 948$
SC-PINN	$8.16 \cdot 10^{-4}$	$7.27 \cdot 10^{-3}$	$t \approx 167$
GF-PSM	$1.6 \cdot 10^{-8}$	$5.4 \cdot 10^{-8}$	$t \approx 0.16$
AD-PSM	$\mathbf{7.61 \cdot 10^{-13}}$	$\mathbf{2.37 \cdot 10^{-12}}$	$t \approx \mathbf{0.07}$
$\mu = 31$	ϵ_1	ϵ_∞	
GF-PSM	$1.09 \cdot 10^{-9}$	$1.45 \cdot 10^{-8}$	$t \approx 2.39$
AD-PSM	$\mathbf{2.25 \cdot 10^{-9}}$	$\mathbf{9.82 \cdot 10^{-9}}$	$t \approx \mathbf{1.07}$

Table 3: Errors for 2D QHO forward problem with $\mu = 21, 31$.

QHO forward Problem

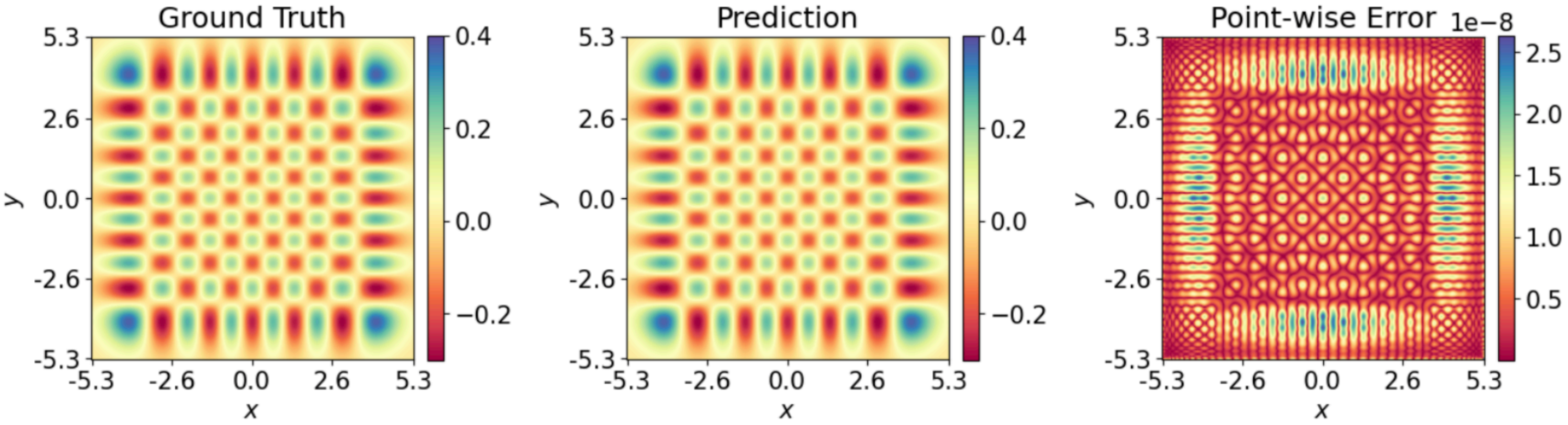


Figure 4: Solution for 2D QHO with $\mu = 31$ on $\Omega' = 5.3\Omega$ due to AD-PSM.

QHO Eigenvalue Inverse Problem

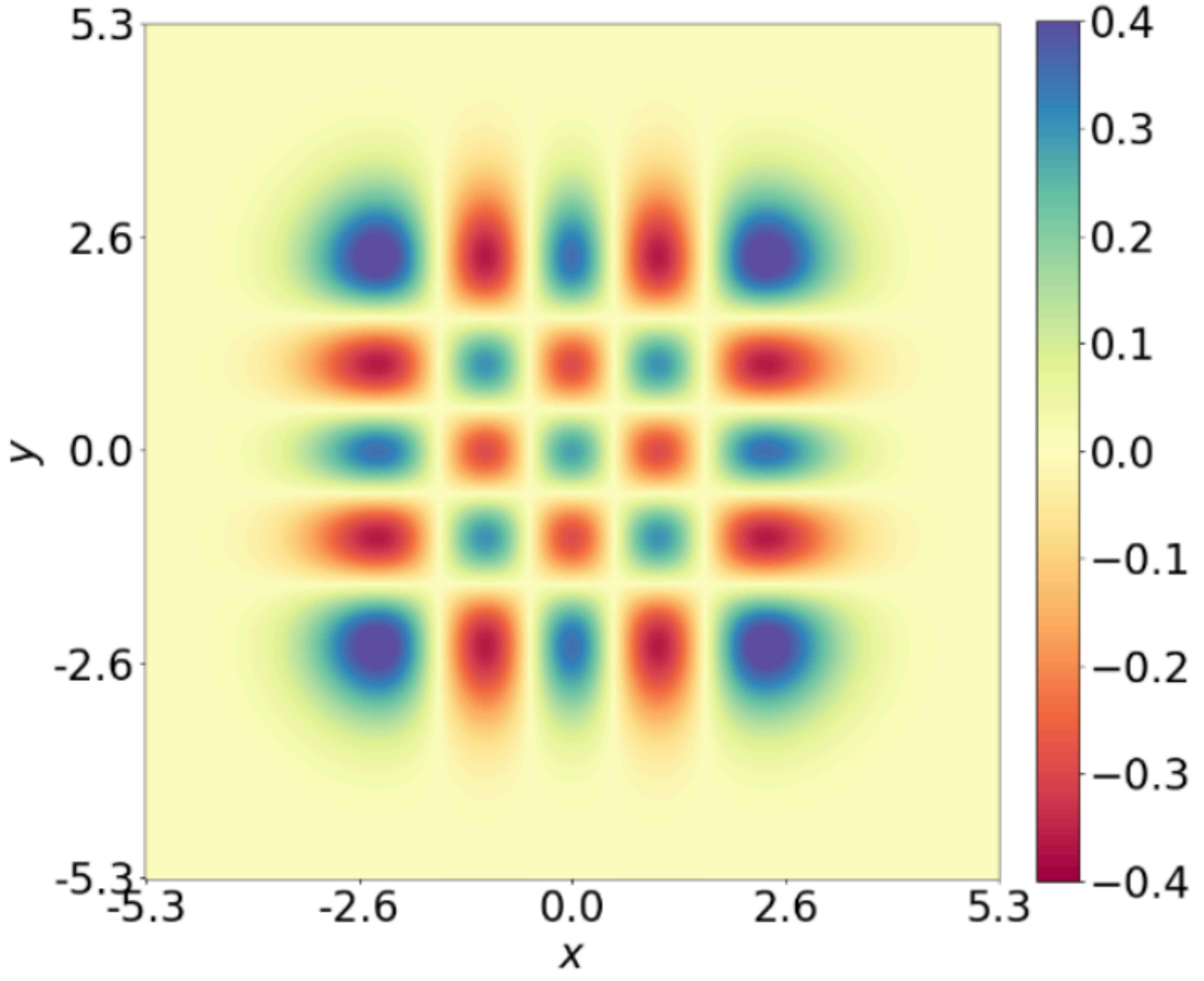


Figure 5: Solution for 2D QHO with $\mu_{gt} = 5$ on $\Omega' = 5.3\Omega$.

	Approximation error		Runtime (s)
	ϵ_μ	ϵ_1	
PINN	6.01	$7.32 \cdot 10^{-2}$	$t \approx 1414$
ID-PINN	$6.21 \cdot 10^{-2}$	$7.51 \cdot 10^{-3}$	$t \approx 1346$
SC-PINN	$2.18 \cdot 10^{-4}$	$5.68 \cdot 10^{-4}$	$t \approx 192$
GF-PSM	$\mathbf{1.50 \cdot 10^{-10}}$	$\mathbf{5.13 \cdot 10^{-9}}$	$t \approx \mathbf{5}$

Table 4: Errors for 2D QHO inverse problem with $\mu_{gt} = 5$

Non-Linear Navier Stokes Problems

We consider the incompressible 2D Navier Stokes equation as an example of a non-linear PDE problem: Let $u = (u_1, u_2)$, $u \in C^2(\Omega, \mathbb{R}^2)$ be the vector velocity field and $p \in C^1(\Omega; \mathbb{R})$ the scalar pressure field the equation becomes:

$$\begin{cases} -\nu \Delta u(x,y) + (u(x,y) \cdot \nabla) u(x,y) + \nabla p(x,y) &= f(x,y) \quad ,\forall (x,y) \in \Omega \\ \nabla \cdot u(x,y) &= 0 \quad ,\forall (x,y) \in \Omega \\ u(x,y) - g(x,y) &= 0 \quad ,\forall (x,y) \in \partial\Omega \,, \end{cases}$$

where

$$f(x,y) = 2\nu\pi^2 (u_1(x,y), u_2(x,y)) + \pi \cos(\pi x) \cos(\pi y)(-u_1(x,y), u_2(x,y))$$
$$+ \pi \sin(\pi x) \sin(\pi y)(u_2, -u_1) + \exp(\pi y)(1, \pi x) \,,$$

$$g(x,y) = [-\sin(\pi x)\cos(\pi y), \cos(\pi x)\sin(\pi y)]^T$$

Non-Linear Navier Stokes Problems

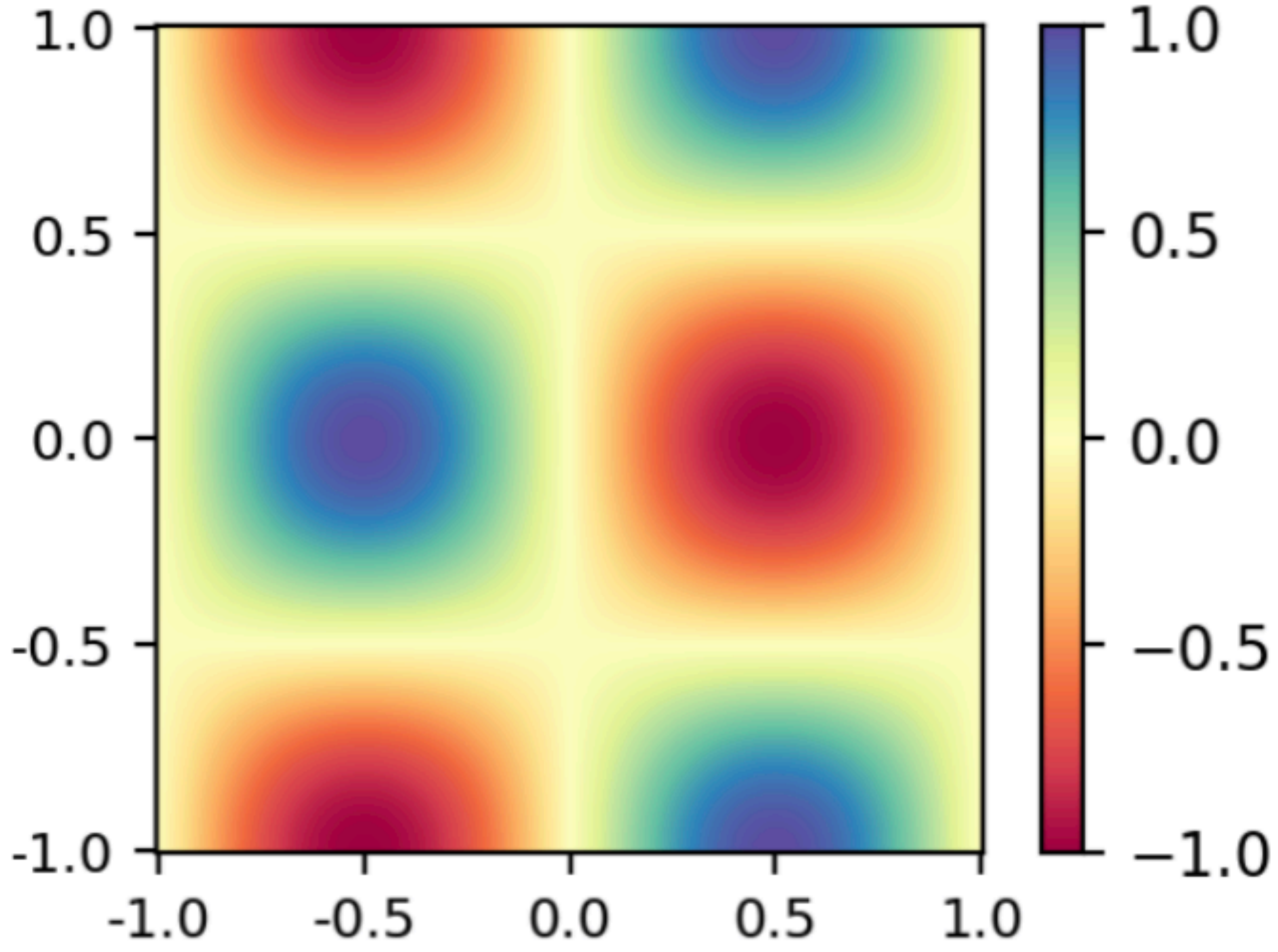


Figure 6: Solution u_1.

	Approximation error		Runtime (s)
Forward Problem	ϵ_1	ϵ_∞	
GF-PSM u_1	$3.31 \cdot 10^{-10}$	$2.35 \cdot 10^{-9}$	$t \approx 405.22$
GF-PSM u_2	$3.28 \cdot 10^{-10}$	$2.35 \cdot 10^{-9}$	$t \approx 405.22$
Inverse Problem	ϵ_ν	ϵ_1	
GF-PSM	$8.9 \cdot 10^{-14}$	$1.07 \cdot 10^{-8}$	$t \approx 395.49$
AD-PSM	$\mathbf{5.9 \cdot 10^{-16}}$	$\mathbf{7.31 \cdot 10^{-15}}$	$t \approx \mathbf{1.75}$

Table 5: Approximation errors

Inverse Problem asks for inferring the viscosity ν and the pressure field !