Physics-Informed Machine Learning in Sound Field Estimation: Fundamentals, state of the art, and challenges

Shoichi Koyama[1] and Mirco Pezzoli[2]

[1]National Institute of Informatics, Tokyo, Japan

[2]Politecnico di Milano, Milan, Italy

NII S.Koyama's Lab
Audio Processing Research Group

ISPL
Image and Sound Processing Lab

About this Webinar

SPECIAL ISSUE ON MODEL-BASED AND
DATA-DRIVEN AUDIO SIGNAL PROCESSING

Shoichi Koyama, Juliano G. C. Ribeiro, Tomohiko Nakamura,
Natsuki Ueno, and Mirco Pezzoli

Physics-Informed Machine Learning for Sound Field Estimation

Fundamentals, state of the art, and challenges

Slides

Paper

Based on our article published in IEEE Signal Processing Magazine

Outline

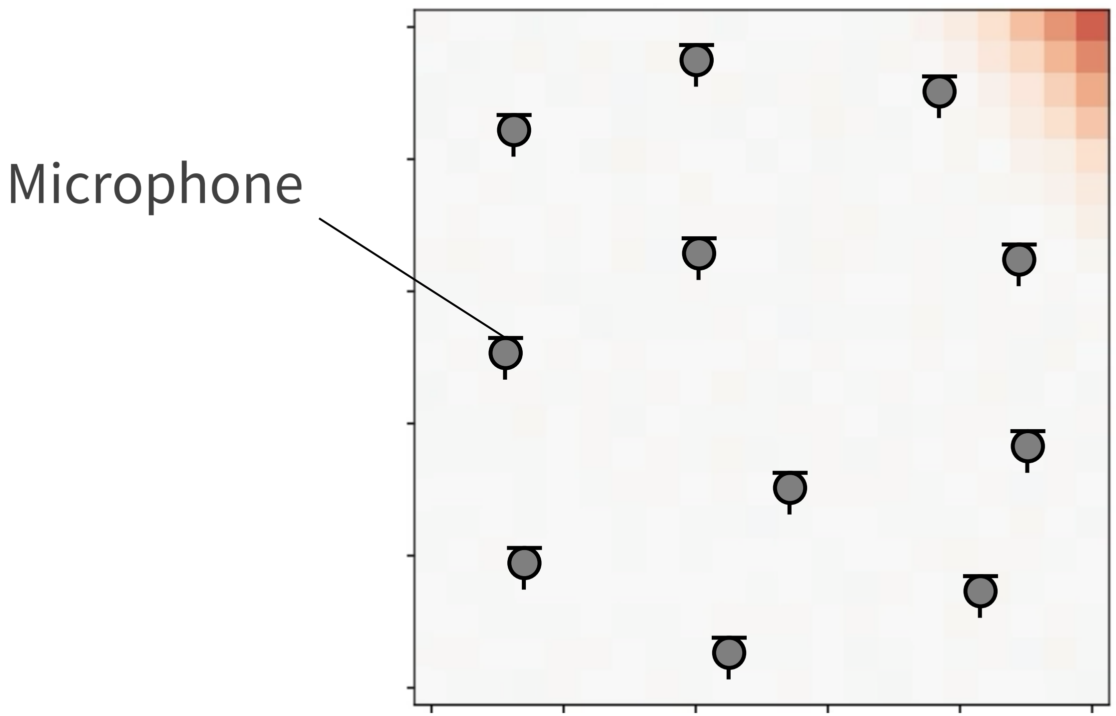
- **What is sound field estimation?**
  - Problem setting
  - Applications
- **Embedding physical properties in interpolation techniques**
  - Basis expansion into element solutions
  - Kernel regression
  - Neural networks incorporating governing PDE
  - PINNs based on implicit neural representation
- **Current studies on sound field estimation based on PIML**
  - Overview of state-of-the-art
- **Outlook**
  - Current limitations and future challenges

Outline

- **What is sound field estimation?**
  - Problem setting
  - Applications
- **Embedding physical properties in interpolation techniques**
  - Basis expansion into element solutions
  - Kernel regression
  - Neural networks incorporating governing PDE
  - PINNs based on implicit neural representation
- **Current studies on sound field estimation based on PIML**
  - Overview of state-of-the-art
- **Outlook**
  - Current limitations and future challenges

What is sound field estimation?

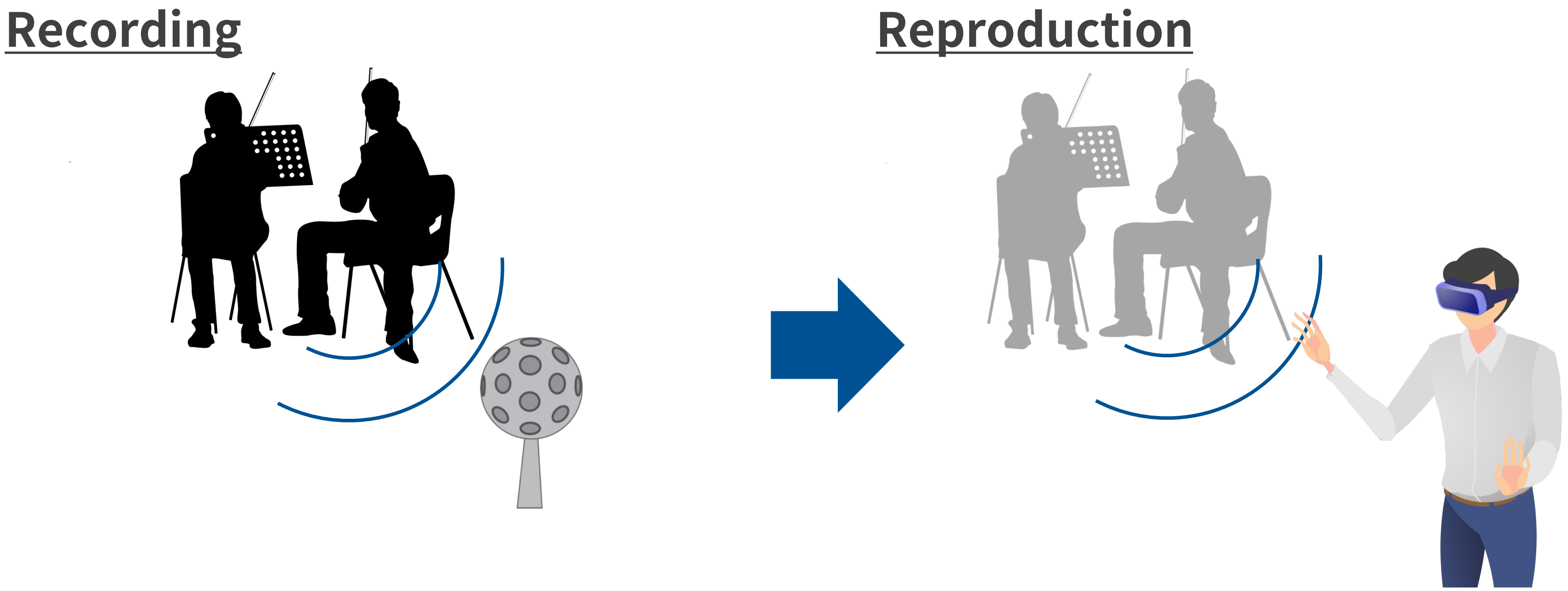
Estimating sound field inside target region using multiple mics



Fundamental technology in various audio processing tasks and has variety of applications

Application #1: Binaural Reproduction

Binaural reproduction from mic array recordings for VR audio



- ➢ Unlike binaural synthesis in VR space, binaural reproduction in real environments requires spatial audio capturing by using multiple mics
- ➢ Required to estimate spatial sound in a wide area to achieve a wide listening area, e.g., 6DoF reproduction

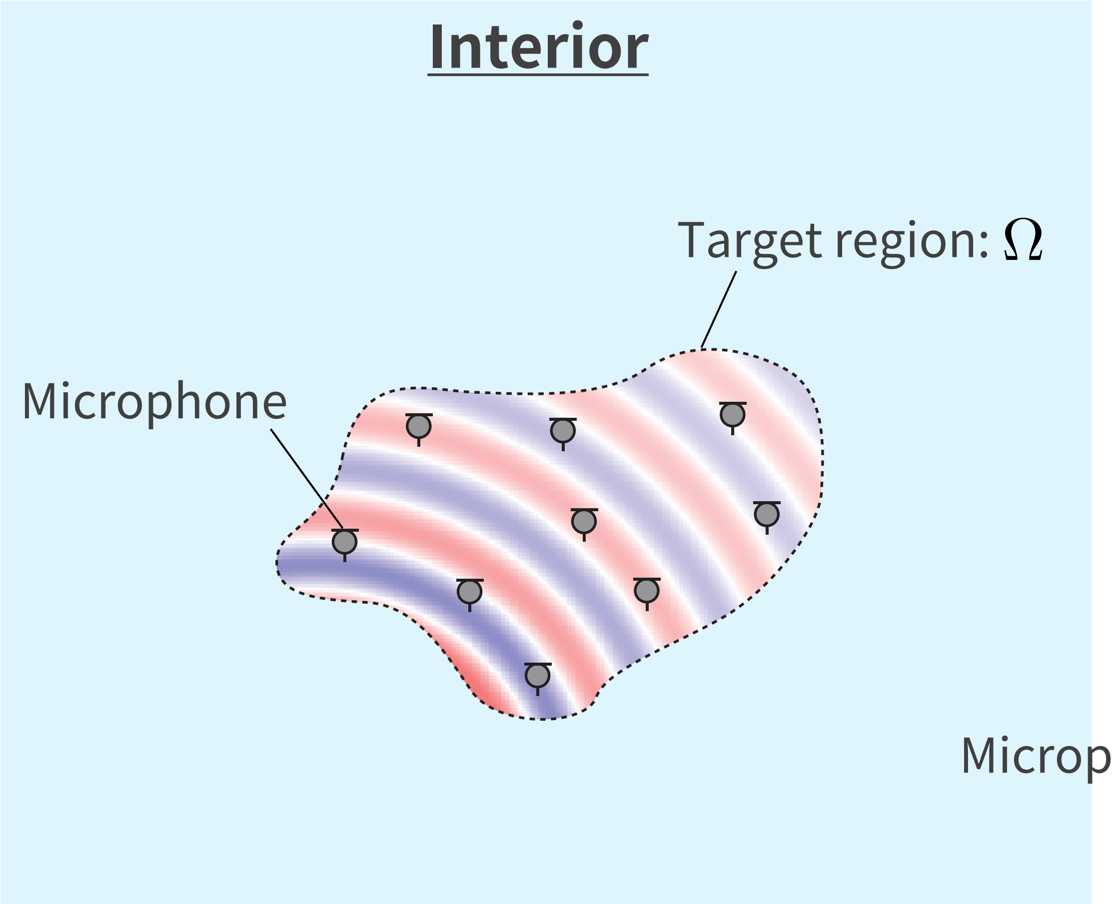
Application #2: Spatial Active Noise Control

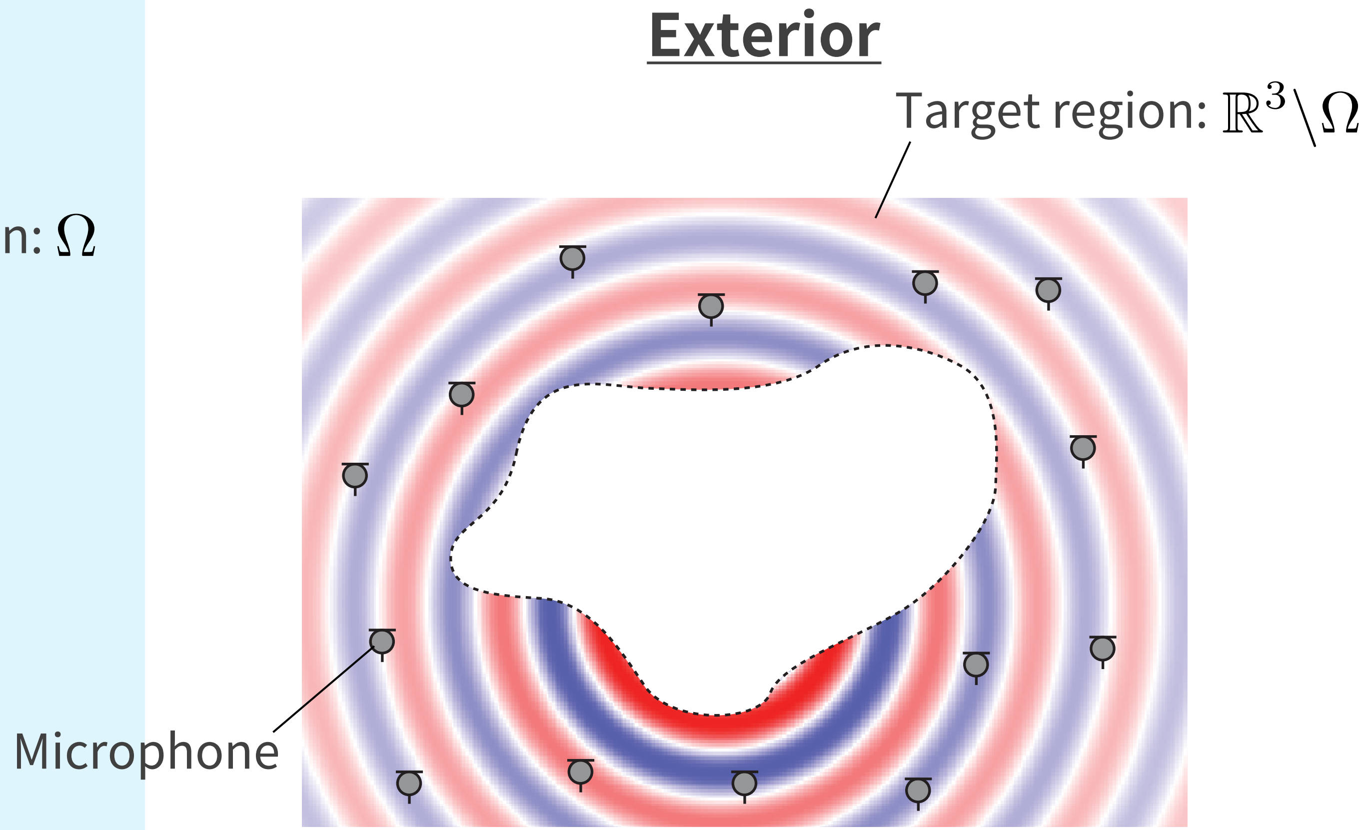
Noise suppression over 3D space by loudspeaker signals

- Active noise control (ANC) aims to cancel noise by using loudspeaker signals, but its effect is limited to local region
- Spatial ANC by estimating spatial sound using multiple mics and synthesizing anti-spatial sound using multiple loudspeakers

Sound field estimation

Interior and exterior sound field estimation

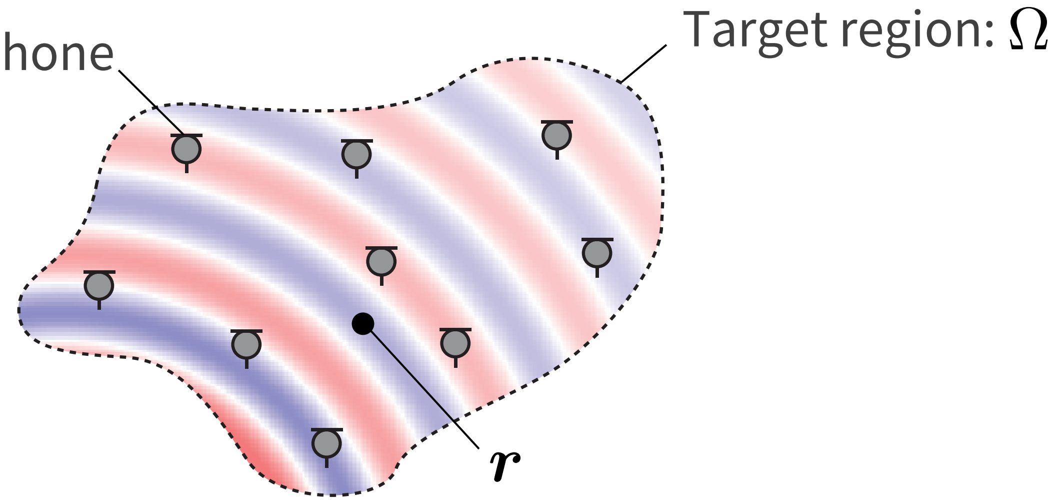




Focusing mainly on estimation in interior free field

Sound field estimation

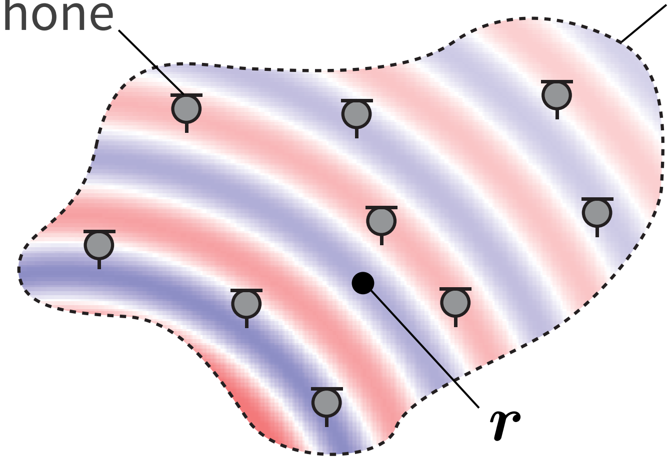
Formulation of sound field estimation problem



Estimate pressure distribution $U(\boldsymbol{r}, t)$ $(\boldsymbol{r} \in \Omega)$ in the time domain or $u(\boldsymbol{r}, \omega)$ in frequency domain with M ominidirectional mics at $\{\boldsymbol{r}_m\}_{m=1}^{M}$

Sound field estimation

Formulation of sound field estimation problem



- Problem to be solved in general interpolation techniques
  - $f \ (U \ \text{or} \ u)$ is represented by model parameters $\boldsymbol{\theta}$

Observation

Regularization term

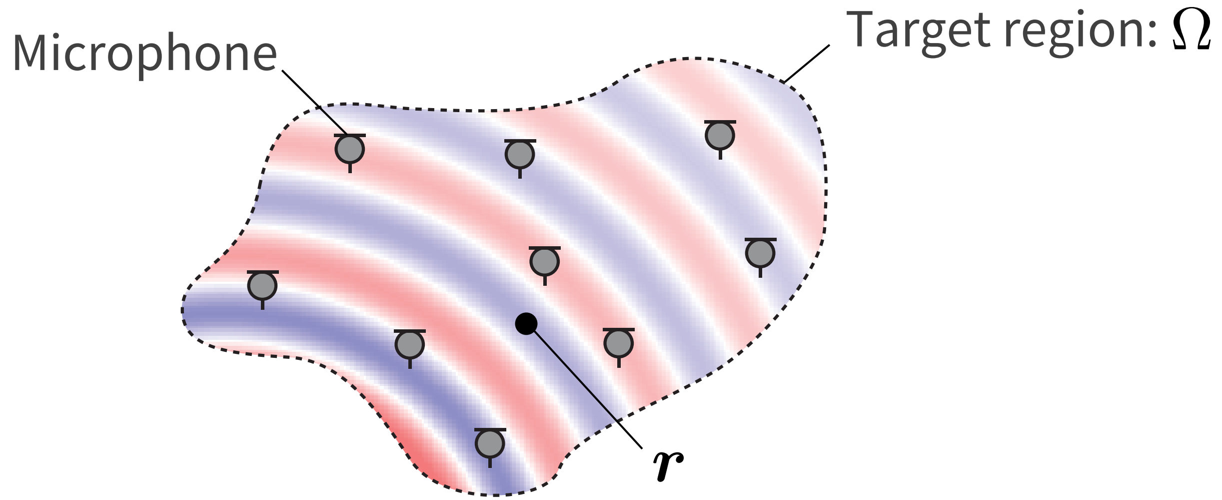
$$\underset{\boldsymbol{\theta}}{\text{minimize}}\, \mathcal{L}(\boldsymbol{y}, f(\{\boldsymbol{x}_i\}_{i=1}^{I}; \boldsymbol{\theta})) + \mathcal{R}(\boldsymbol{\theta})$$

Loss term

Samples in space/time/freq

Sound field estimation

Formulation of sound field estimation problem



- Problem to be solved in general interpolation techniques
  - $f \ (U \text{ or } u)$ is represented by model parameters $\boldsymbol{\theta}$

Squared ℓ_2-norm penalty

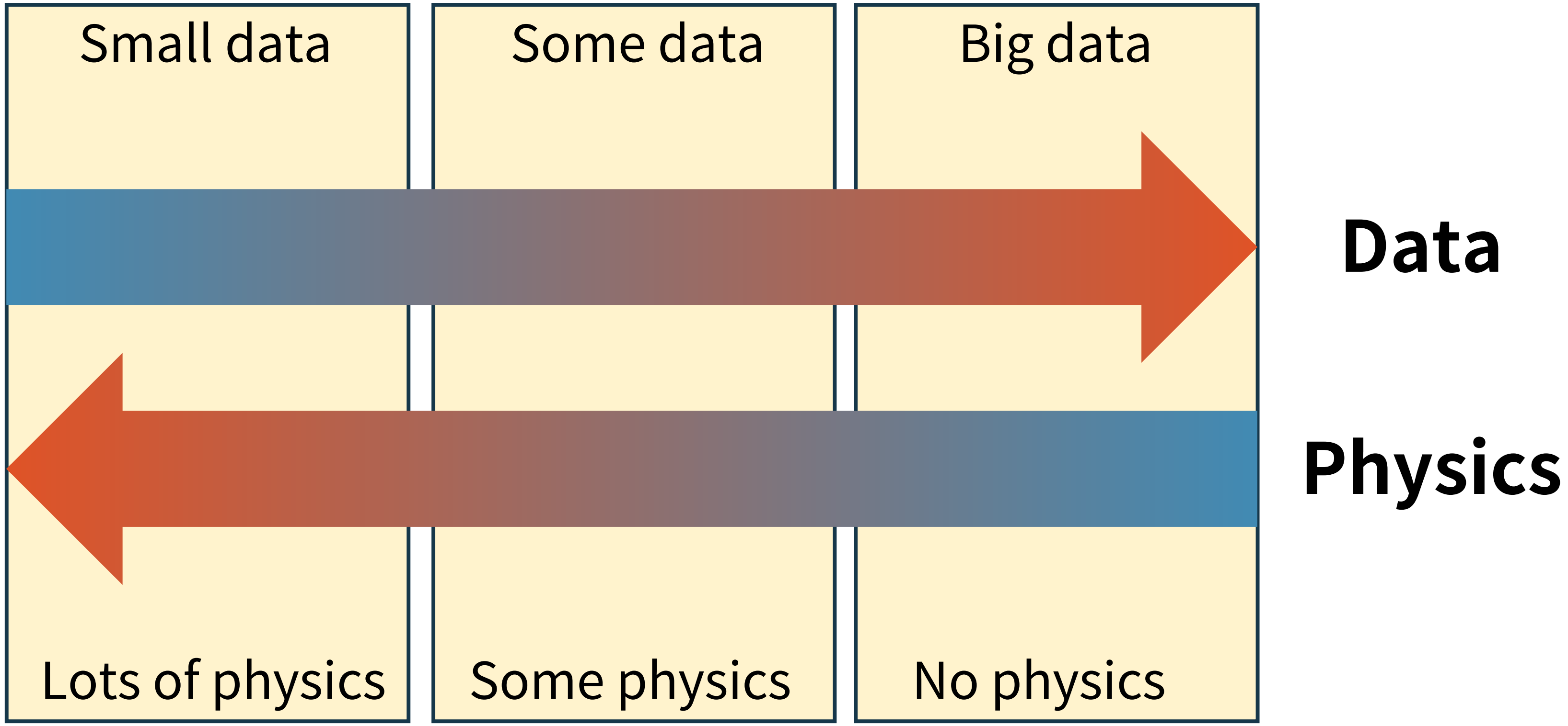
$$\underset{\boldsymbol{\theta}}{\text{minimize}} \left\| \boldsymbol{y} - f(\{\boldsymbol{x}_i\}_{i=1}^{I}; \boldsymbol{\theta}) \right\|^2 + \lambda \|\boldsymbol{\theta}\|^2$$

Squared error loss

EMBEDDING PHYSICAL PROPERTIES IN INTERPOLATION TECHNIQUES

Embedding physical properties in interpolation techniques

Purely data-driven approaches may suffer from overfitting



[Karniadakis+ 2021]

Physical properties will be useful prior information in sound field estimation

Embedding physical properties in interpolation techniques

What kind of physical properties can be embedded?

- Function to be estimated should satisfy governing PDE
  - **Wave equation** in time domain

$$\left(\nabla_{\boldsymbol{r}}^2 - \frac{1}{c^2}\frac{\partial^2}{\partial t^2}\right) U(\boldsymbol{r}, t) = 0$$

  - **Helmholtz equation** in freq domain

$$\left(\nabla_{\boldsymbol{r}}^2 + k^2\right) u(\boldsymbol{r}, \omega) = 0$$

➡ **Techniques incorporating constraints on the governing PDEs are introduced**

Basis expansion into element solutions

Linear combination of finite number of basis functions

- Function f is modeled by basis functions $\{\varphi_l(\boldsymbol{x})\}_{l=1}^{L}$ and their weights $\{\gamma_l\}_{l=1}^{L}$

$$f(\boldsymbol{x};\boldsymbol{\gamma}) = \sum_{l=1}^{L} \gamma_l \varphi_l(\boldsymbol{x})$$
$$= \boldsymbol{\varphi}(\boldsymbol{x})^{\mathsf{T}}\boldsymbol{\gamma}$$

$$\boldsymbol{\varphi} = [\varphi_1, \dots, \varphi_L]^{\mathsf{T}}$$

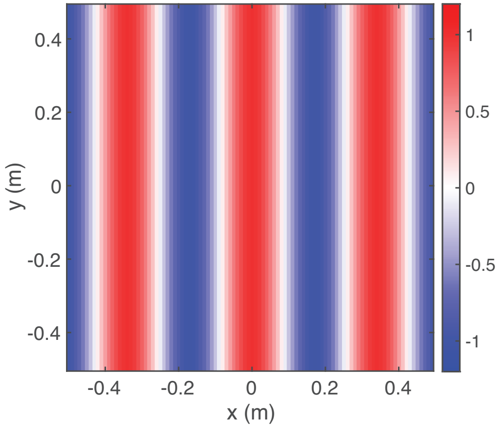
$$\boldsymbol{\gamma} = [\gamma_1, \dots, \gamma_L]^{\mathsf{T}}$$

- **Basis functions as element solutions of wave/Helmholtz eq** [Williams+ 1999, Colton&Kress 2019]
  - Plane wave expansion (Herglotz wave function)
  - Spherical wave function expansion
  - Equivalent source distribution (single-layer potential)

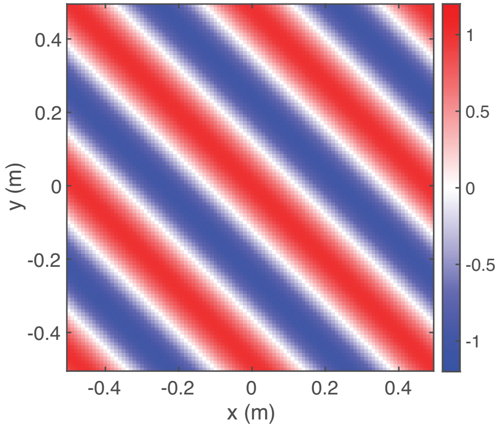
➢ Plane wave expansion (or Herglotz wave function)

Plane wave arrival direction

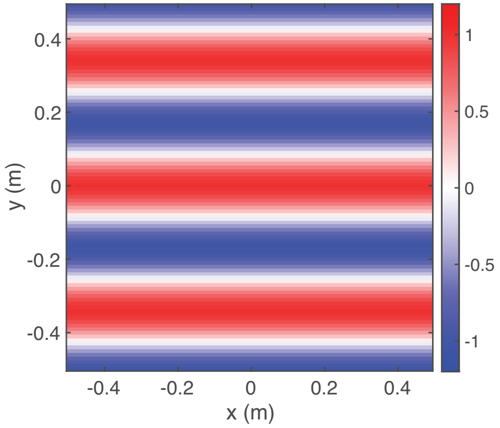
$$u(\boldsymbol{r},\omega)=\int_{\mathbb{S}_2}\tilde{u}(\boldsymbol{\eta},\omega)\mathrm{e}^{\mathrm{j}k\langle\boldsymbol{\eta},\boldsymbol{r}\rangle}\mathrm{d}\boldsymbol{\eta}$$



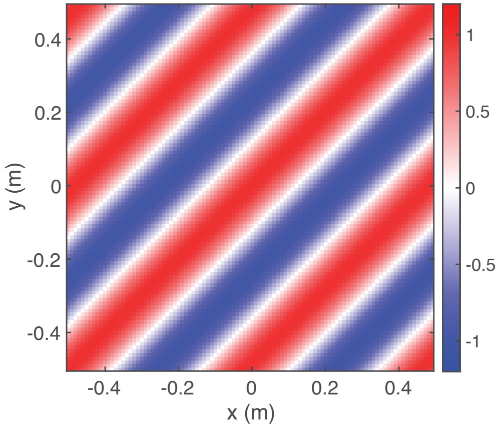
$\boldsymbol{\eta}=[1.0,0.0,0.0]^\mathsf{T}$



$\boldsymbol{\eta}=[0.7,0.7,0.0]^\mathsf{T}$



$\boldsymbol{\eta}=[0.0,1.0,0.0]^\mathsf{T}$



$\boldsymbol{\eta}=[-0.7,0.7,0.0]^\mathsf{T}$

Basis expansion into element solutions

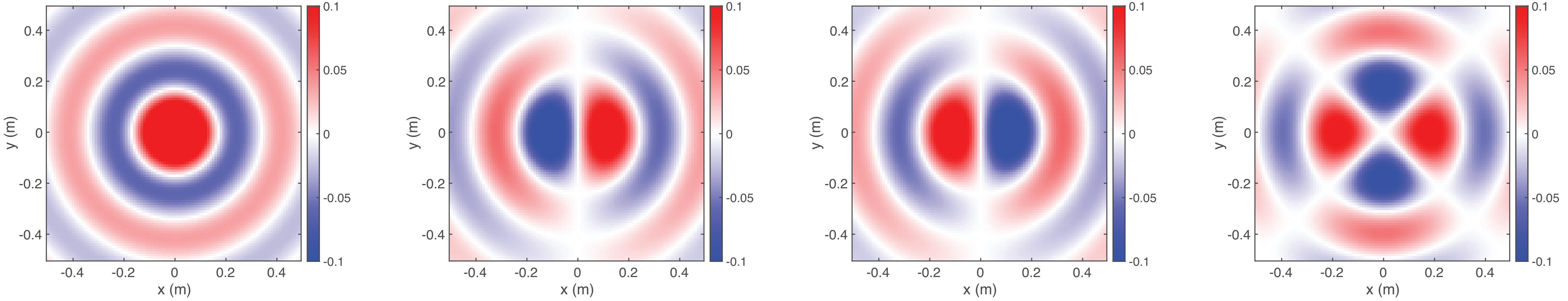
➢ Spherical wave function expansion

Expansion center

Spherical harmonic function

$$u(\boldsymbol{r},\omega) = \sum_{\nu=0}^{\infty} \sum_{\mu=-\infty}^{\infty} \mathring{u}_{\nu,\mu}(\boldsymbol{r}_{\mathrm{o}},\omega) j_{\nu}(k\|\boldsymbol{r}-\boldsymbol{r}_{\mathrm{o}}\|) Y_{\nu,\mu}\left(\frac{\boldsymbol{r}-\boldsymbol{r}_{\mathrm{o}}}{\|\boldsymbol{r}-\boldsymbol{r}_{\mathrm{o}}\|}\right)$$

Spherical Bessel function

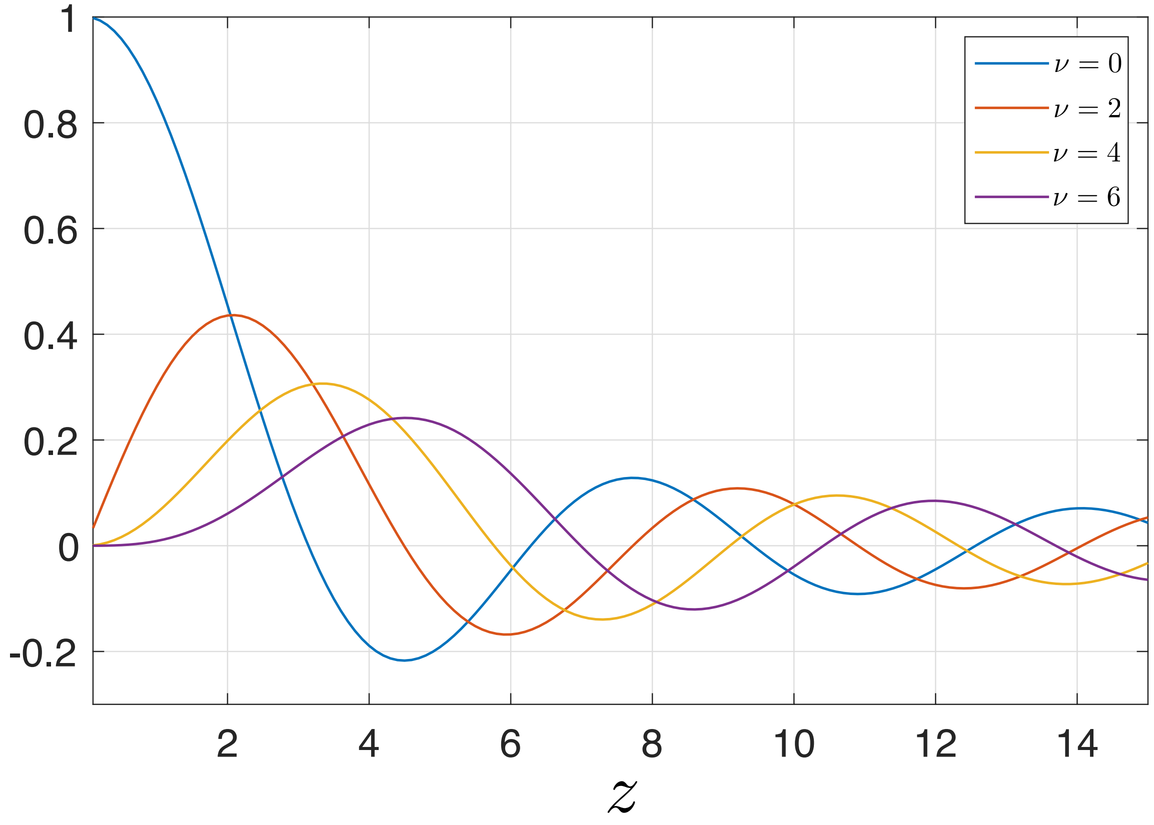


$(\nu,\mu)=(0,0)$ $(\nu,\mu)=(1,1)$ $(\nu,\mu)=(1,-1)$ $(\nu,\mu)=(2,2)$

Basis expansion into element solutions

➢ Spherical Bessel function

$$j_\nu(z) = \sqrt{\frac{\pi}{2z}}\, \underbrace{J_{\nu+1/2}}_{\text{Bessel function}}(z)$$

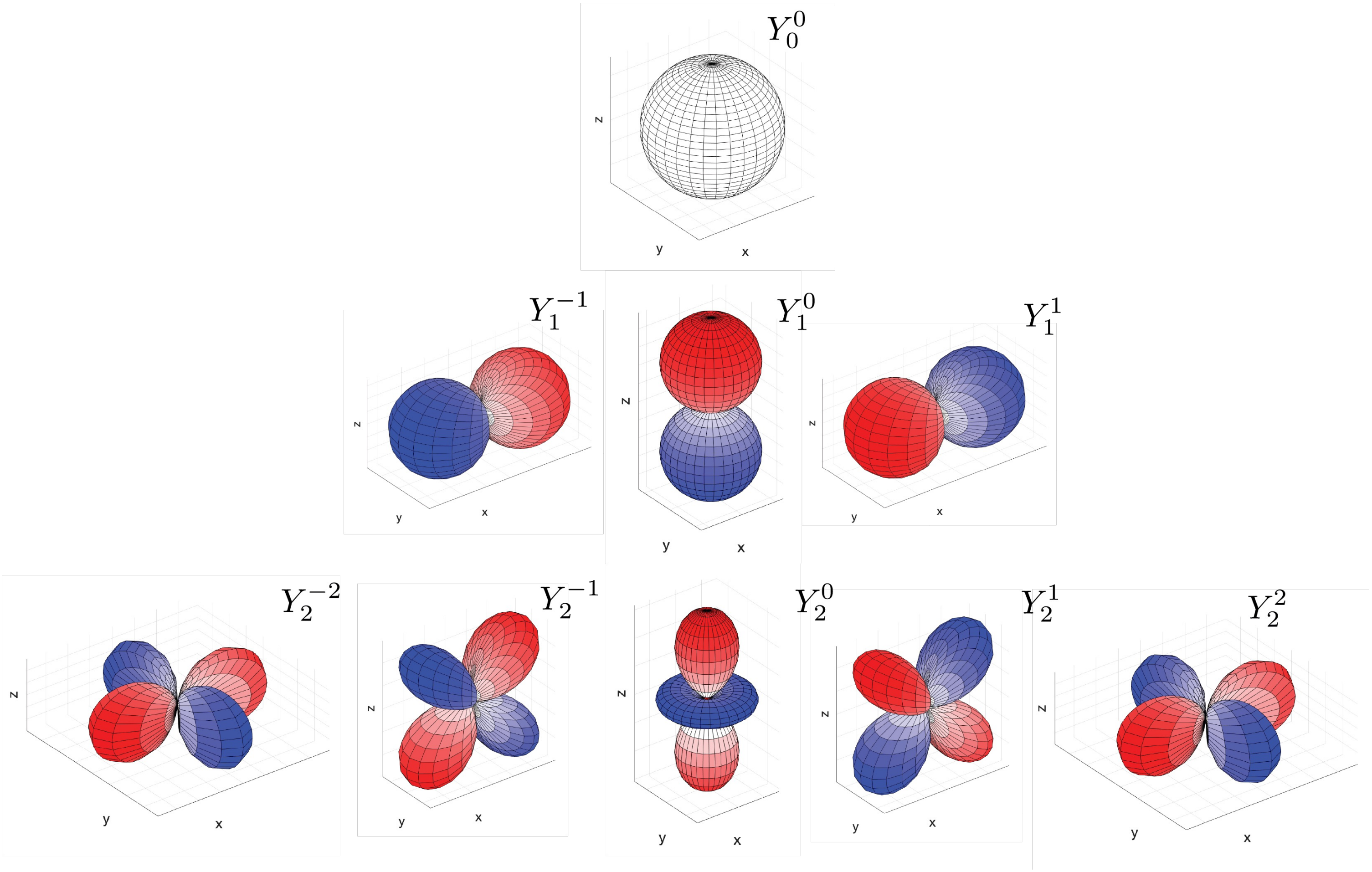


Basis expansion into element solutions

- Spherical harmonic function

Associated Legendre function

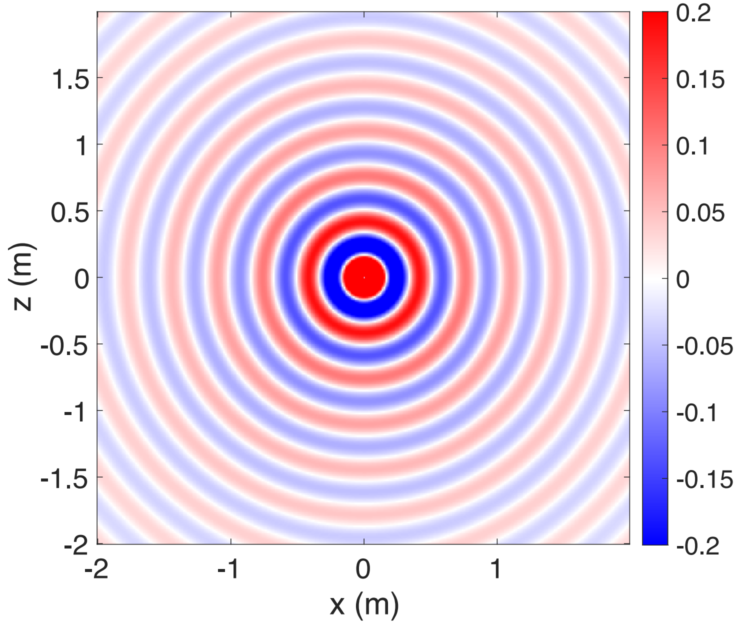
$$Y_{\nu,\mu}(\theta,\phi) = \sqrt{\frac{(2\nu+1)}{4\pi}\frac{(\nu-\mu)!}{(\nu+\mu)!}}P_\nu^\mu(\cos\theta)\mathrm{e}^{\mathrm{j}\mu\phi}$$

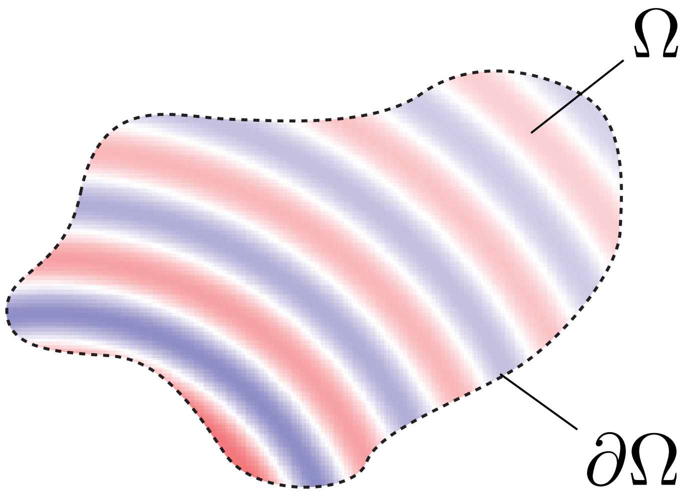


➢ Equivalent source distribution (or single layer potential)

Point source

$$u(\boldsymbol{r},\omega) = \int_{\partial\Omega} \breve{u}(\boldsymbol{r}',\omega)\frac{\mathrm{e}^{-\mathrm{j}k(\boldsymbol{r}-\boldsymbol{r}')}}{4\pi\|\boldsymbol{r}-\boldsymbol{r}'\|}\mathrm{d}\boldsymbol{r}'$$





Basis expansion into element solutions

- Linear regression with finite-dimensional basis expansion
  - Regularized least squares solution of expansion coefs

$$
\begin{aligned}
\hat{\gamma} &= \arg\min_{\boldsymbol{\gamma}} \|\boldsymbol{y} - \boldsymbol{\Phi}\boldsymbol{\gamma}\|^2 + \lambda\|\boldsymbol{\gamma}\|^2 \\
&= \left(\boldsymbol{\Phi}^{\mathsf{H}}\boldsymbol{\Phi} + \lambda\boldsymbol{I}\right)^{-1}\boldsymbol{\Phi}^{\mathsf{H}}\boldsymbol{y}
\end{aligned}
$$

$$\boldsymbol{\Phi} = [\boldsymbol{\varphi}(\boldsymbol{x}_1), \ldots, \boldsymbol{\varphi}(\boldsymbol{x}_I)]^{\mathsf{T}}$$

  - Estimate the function f

$$\hat{f}(\boldsymbol{x}; \hat{\gamma}) = \boldsymbol{\varphi}(\boldsymbol{x})^{\mathsf{T}}\hat{\gamma}$$

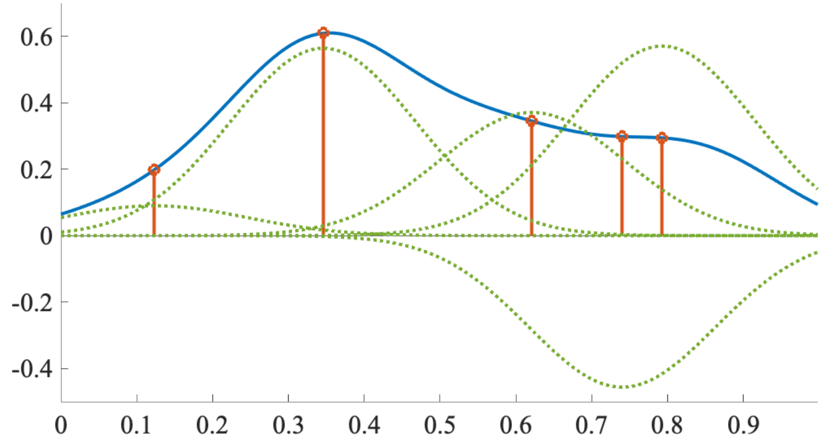
Number of basis functions (and expansion center for spherical wave function expansion) have to be properly set

Kernel regression with constraint of governing PDE

- f is represented by weighted sum of kernel function κ

$$f(\boldsymbol{x};\boldsymbol{\alpha}) = \sum_{i=1}^{I} \alpha_i \kappa(\boldsymbol{x}, \boldsymbol{x}_i)$$
$$= \boldsymbol{\kappa}(\boldsymbol{x})^\mathsf{T} \boldsymbol{\alpha}$$

$$\boxed{\boldsymbol{\kappa}(\boldsymbol{x}) = [\kappa(\boldsymbol{x}, \boldsymbol{x}_1), \ldots, \kappa(\boldsymbol{x}, \boldsymbol{x}_I)]^\mathsf{T}} \quad \boxed{\boldsymbol{\alpha} = [\alpha_1, \ldots, \alpha_I]^\mathsf{T}}$$



- Kernel function κ is a similarity function expressed as innter product on some functional space $\mathscr{H}$

$$\kappa(\boldsymbol{x}, \boldsymbol{x}') = \langle \boldsymbol{\varphi}(\boldsymbol{x}), \boldsymbol{\varphi}(\boldsymbol{x}') \rangle_{\mathscr{H}}$$

➡ $\boldsymbol{\varphi}(\boldsymbol{x})$ **can be infinite-dimensional or κ can be directly designed**

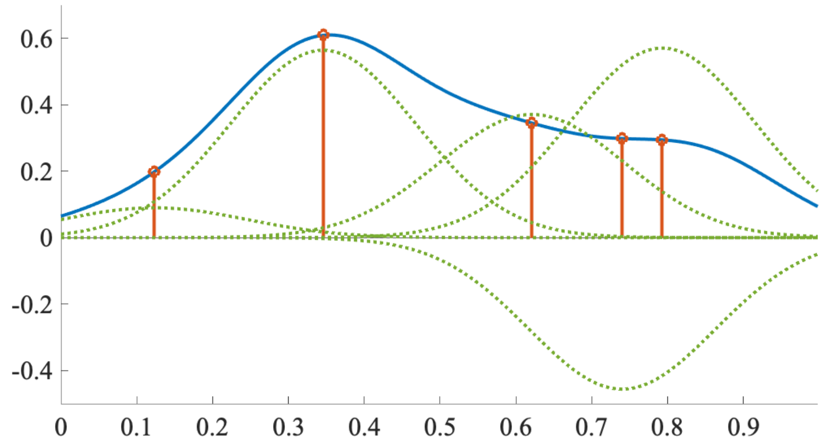
Kernel regression with constraint of governing PDE

- In kernel ridge regression, $\boldsymbol{\alpha}$ is obtained as

$$\hat{\boldsymbol{\alpha}} = (\boldsymbol{K} + \lambda \boldsymbol{I})^{-1} \boldsymbol{y}$$

with Gram matrix defined as

$$\boldsymbol{K} = \begin{bmatrix} \kappa(\boldsymbol{x}_1, \boldsymbol{x}_1) & \dots & \kappa(\boldsymbol{x}_1, \boldsymbol{x}_I) \\ \vdots & \ddots & \vdots \\ \kappa(\boldsymbol{x}_I, \boldsymbol{x}_1) & \dots & \kappa(\boldsymbol{x}_I, \boldsymbol{x}_I) \end{bmatrix}$$



- Estimate the function

$$\hat{f}(\boldsymbol{x}; \hat{\boldsymbol{\alpha}}) = \boldsymbol{\kappa}(\boldsymbol{x})^{\mathsf{T}} \hat{\boldsymbol{\alpha}}$$

➡ Function space $\mathscr{H}$, which also defines κ, must be properly defined

Kernel regression with constraint of governing PDE

Kernel function to constrain the solution to satisfy Helmholtz eq

- Inner product and norm over $\mathscr{H}$ are defined by plane wave expansion with positive directional weighting w [Ueno+ 2021]

$$\langle u_1, u_2 \rangle_{\mathscr{H}} = 4\pi \int_{\mathbb{S}_2} \underline{\frac{1}{w(\boldsymbol{\eta})}} \tilde{u}_1(\boldsymbol{\eta})^* \tilde{u}_2(\boldsymbol{\eta}) \mathrm{d}\boldsymbol{\eta}$$

$$\|u\|_{\mathscr{H}} = \sqrt{\langle u, u \rangle_{\mathscr{H}}}$$

Directional weighting w is designed to incorporate prior knowledge of sound field directivity

Kernel regression with constraint of governing PDE

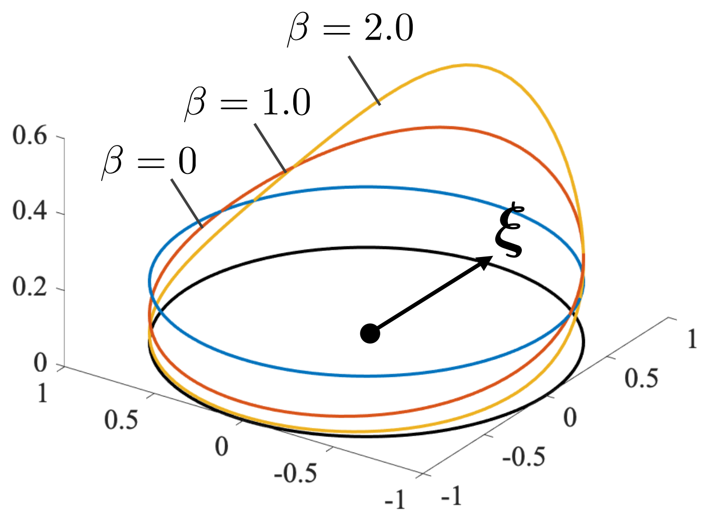
Kernel function to constrain the solution to satisfy Helmholtz eq

- Kernel function when w is defined by using von Mises–Fisher distribution

$$w(\boldsymbol{\eta}) = \frac{1}{C(\beta)} \mathrm{e}^{\beta \langle \boldsymbol{\eta}, \boldsymbol{\xi} \rangle}$$

$$\kappa(\boldsymbol{r}_1, \boldsymbol{r}_2) = \frac{1}{C(\beta)} j_0 \left(\sqrt{(k\boldsymbol{r}_{12} - \mathrm{j}\beta\boldsymbol{\xi})^{\mathsf{T}} (k\boldsymbol{r}_{12} - \mathrm{j}\beta\boldsymbol{\xi})} \right)$$

with $\boldsymbol{r}_{12} = \boldsymbol{r}_2 - \boldsymbol{r}_1$

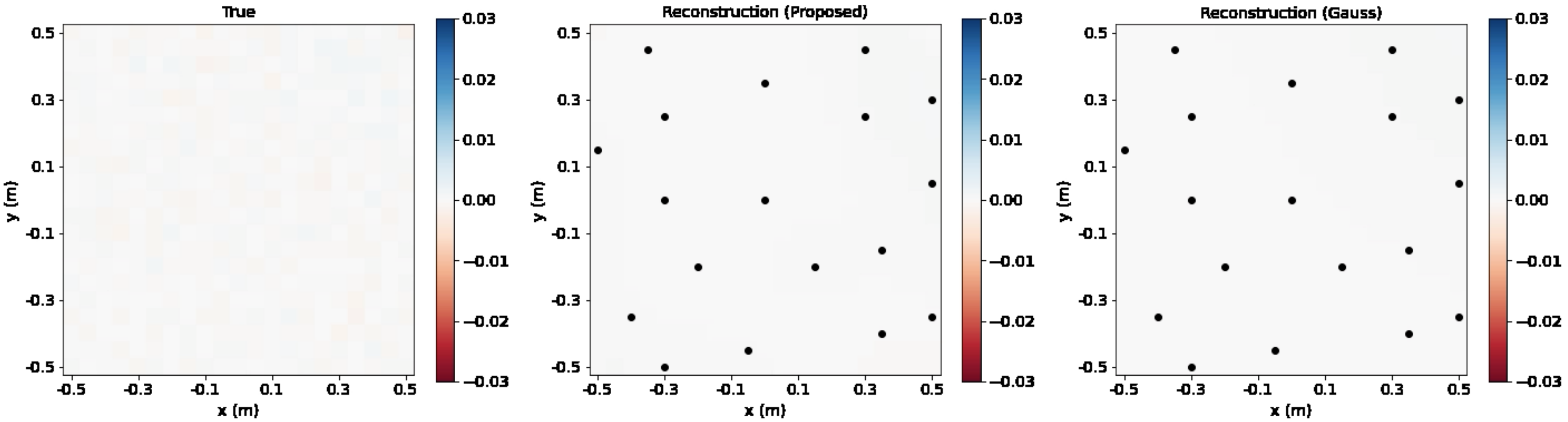


- When no prior information, i.e., uniform weight $w(\boldsymbol{\eta}) = 1$,

$$\kappa(\boldsymbol{r}_1, \boldsymbol{r}_2) = j_0(k\|\boldsymbol{r}_2 - \boldsymbol{r}_1\|)$$

- Experimental results using real data from MeshRIR dataset [Koyama+ 2021]
  - Reconstructing pulse signal from single loudspeaker w/ 18 mic

Ground Truth | **Kernel Interpolation (Spherical Bessel)** | **Kernel Interpolation (Gaussian)**



(Black dots indicate mic positions)

Neural Network-based sound field estimation

Why NNs in sound field estimation?

- High representational power
  - Solution space in basis expansion and kernel regression is highly constrained
  - High adaptability to the target acoustic environment can be expected by using NNs

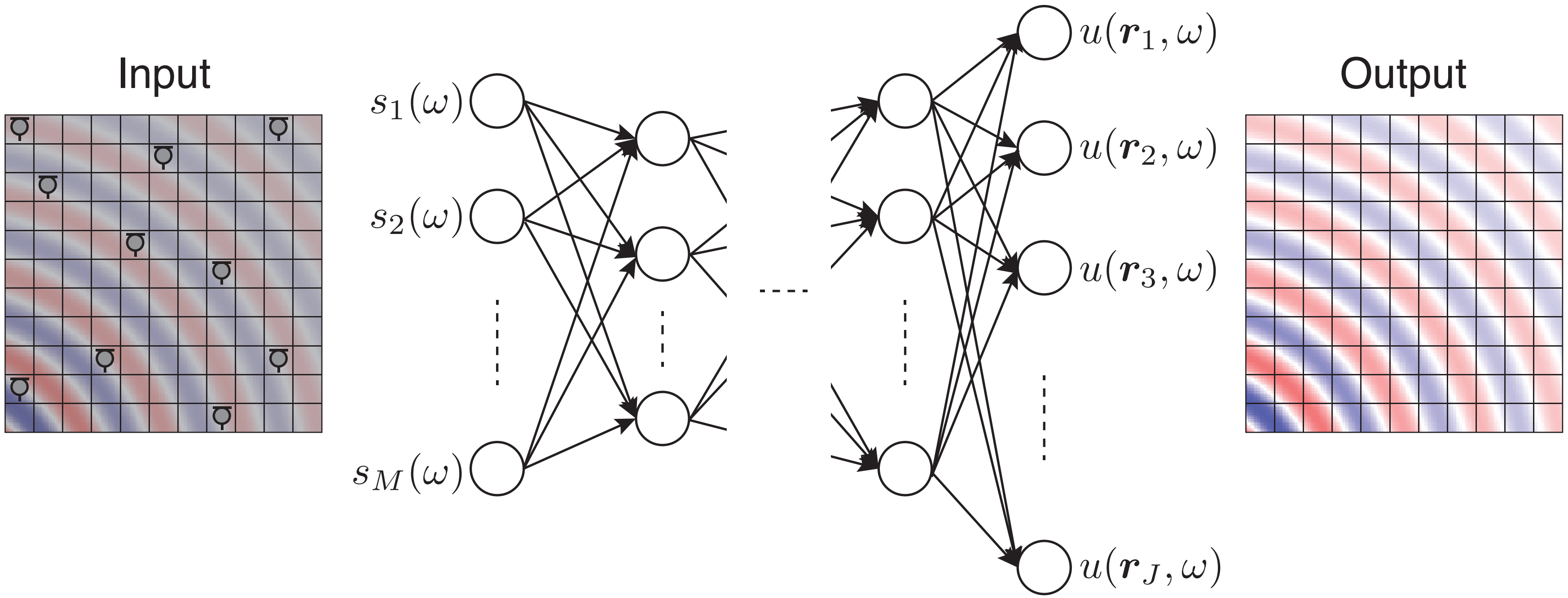
- From snapshot-based (**unsupervised**) to learning-based (**supervised**)
  - Basically, linear and kernel regressions use only a snapshot observation
  - Properties of the target acoustic environment can be learned from training data

➡ **Highly accurate estimation can be expected, especially when the number of mics is extremely small**

Feedforward NNs incorporating governing PDEs

- Regression by feedforward NNs
  - Target output is discretized as $\boldsymbol{t} = [f(\boldsymbol{x}_1), \ldots, f(\boldsymbol{x}_J)]^\mathsf{T}$
  - NN with input $\boldsymbol{y}$ and output $\boldsymbol{g}(\boldsymbol{y}; \boldsymbol{\theta}_{\mathrm{NN}})$ is designed with NN params $\boldsymbol{\theta}_{\mathrm{NN}}$
  - NN is trained using a pair of datasets $\{(\boldsymbol{y}_d, \boldsymbol{t}_d)\}_{d=1}^{D}$ to minimize the loss, e.g.,

$$\mathcal{J}(\boldsymbol{\theta}_{\mathrm{NN}}) = \sum_{d=1}^{D} \|\boldsymbol{t}_d - \boldsymbol{g}(\boldsymbol{y}_d; \boldsymbol{\theta}_{\mathrm{NN}})\|^2$$



Feedforward NNs incorporating governing PDEs

How to embed governing PDEs to feedforward NNs?

➢ Estimating weights of basis expansion using NNs

- Train a NN estimating weights of basis expansion
- Continuous function can be reconstructed by using estimated expansion coefs
- Can be regarded as **physics-constrained neural network (PCNN)** [Karakonstantis+ 2023, Lobato+ 2024]

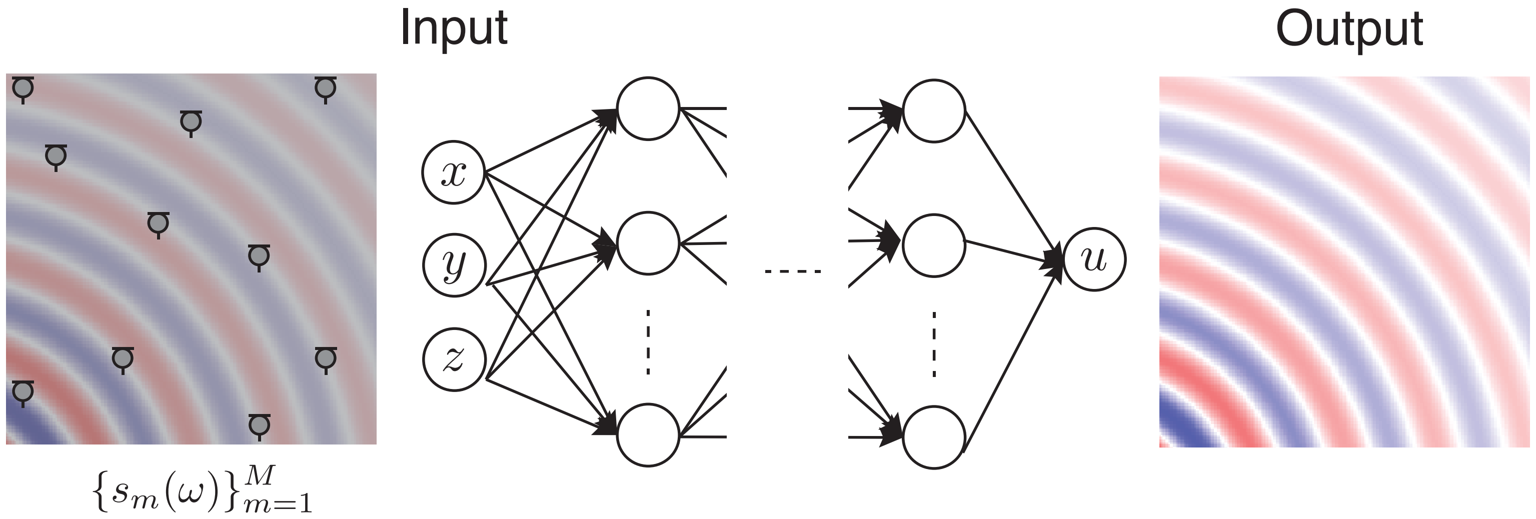
➢ Incorporating (approximate) PDE loss

- Loss function evaluating deviation from governing PDEs: **PDE loss**
- Because of discrete output values, PDE loss is computed by finite difference or interpolation
- In [Shigemi+ 2022], **physics-informed convolutional neural network (PICNN)** using bi-cubic spline interpolation is proposed

PINNs based on implicit neural representation

- **Implicit neural representation** [Sitzmann+ 2020]
  - NNs are used to implicity represent a continuous function f
  - NN with input $\boldsymbol{x}$ and output $g(\boldsymbol{x}; \boldsymbol{\theta}_{\mathrm{NN}})$ is designed with NN params $\boldsymbol{\theta}_{\mathrm{NN}}$
  - NN is trained for approximaging $f(\boldsymbol{x})$ by using training data $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^{I}$

$$\mathcal{J}_{\mathrm{INR}}(\boldsymbol{\theta}_{\mathrm{NN}}) = \sum_{i=1}^{I} |y_i - g(\boldsymbol{x}_i; \boldsymbol{\theta}_{\mathrm{NN}})|^2$$

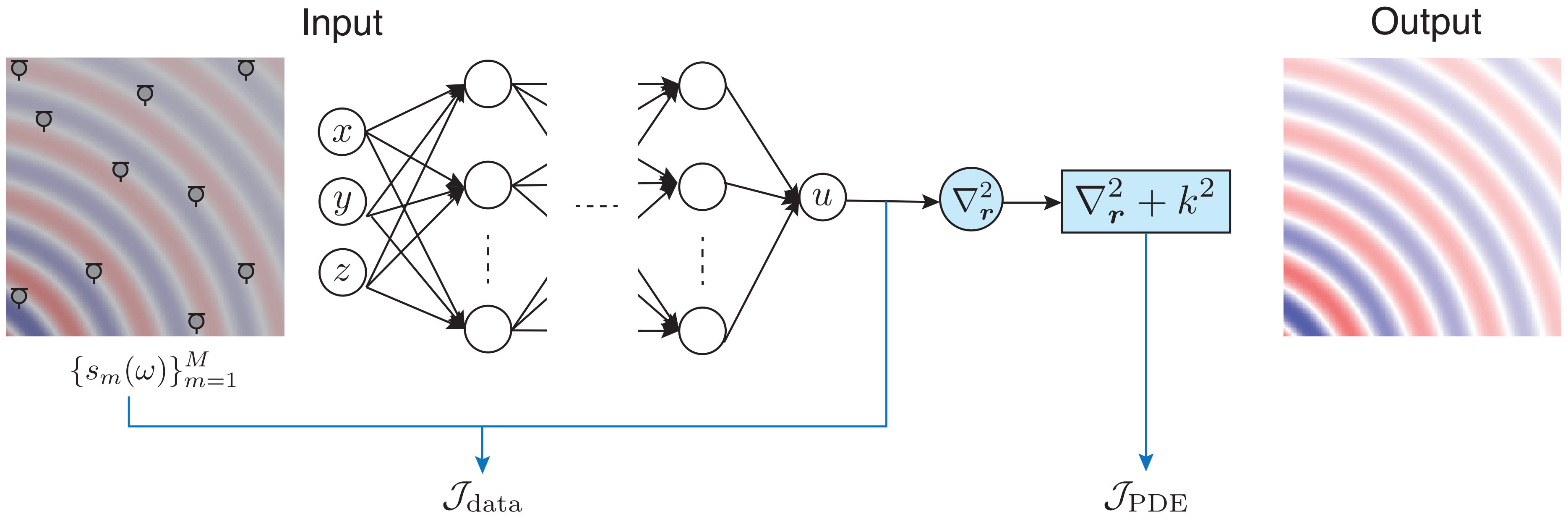


PINNs based on implicit neural representation

- **Physics-informed neural network (PINN)** [Raissi+ 2019]
  - Implicit neural representation allows incorporating constraints on g including its (partial) derivatives in loss function

$$\mathcal{J}_{\mathrm{INR}}(\boldsymbol{\theta}_{\mathrm{NN}}) = \sum_{i=1}^{I} |y_i - g(\boldsymbol{x}_i; \boldsymbol{\theta}_{\mathrm{NN}})|^2 + \epsilon \sum_{n=1}^{N} \left| H\left(g(\boldsymbol{x}_n), \nabla_{\boldsymbol{x}} g(\boldsymbol{x}_n), \nabla_{\boldsymbol{x}}^2 g(\boldsymbol{x}_n), \ldots\right) \right|^2$$

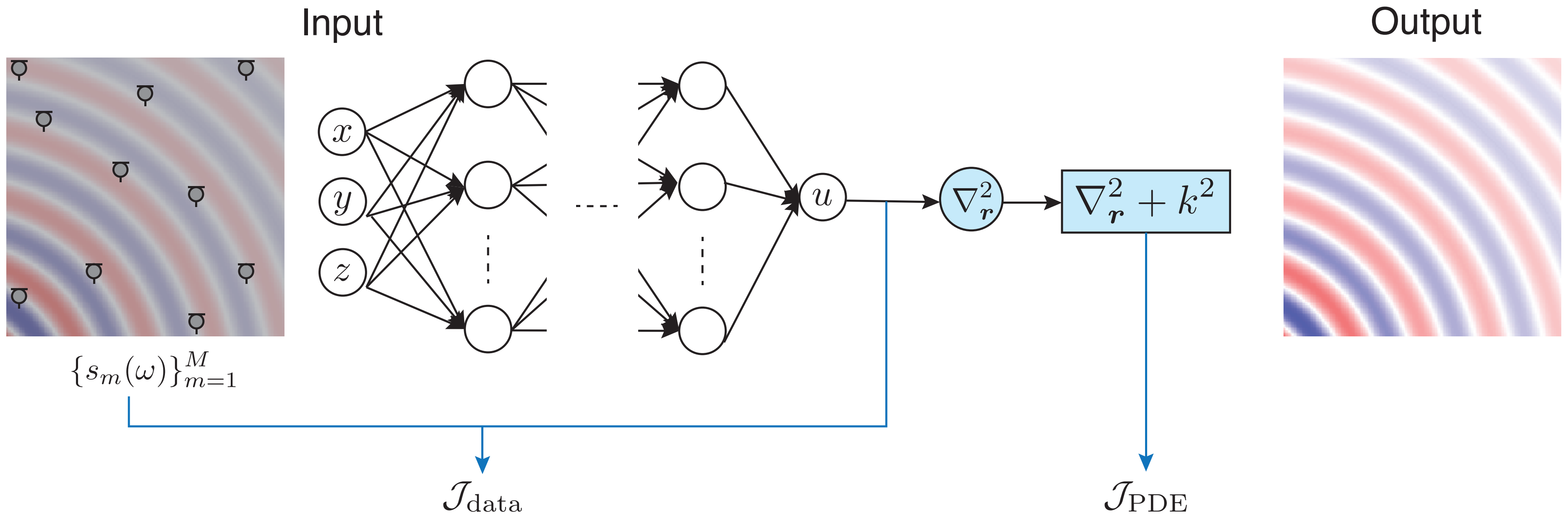
Usually computed by automatic differentiation



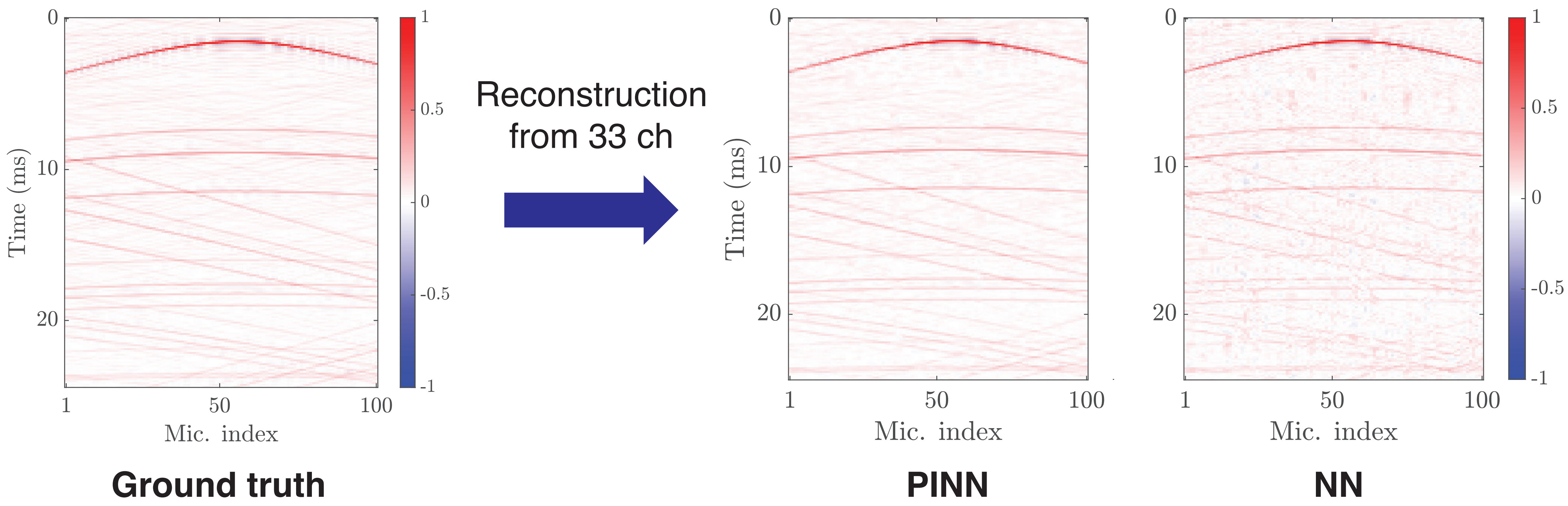
- **Physics-informed neural network (PINN)** [Raissi+ 2019]
  - Case when estimating function approximately satisfying Helmhotz eq

$$\mathcal{J}_{\mathrm{PINN}}(\boldsymbol{\theta}_{\mathrm{NN}}) = \underbrace{\sum_{m=1}^{M} |s_m(\omega) - g(\boldsymbol{r}_m; \boldsymbol{\theta}_{\mathrm{NN}})|^2}_{\mathcal{J}_{\mathrm{data}}} + \epsilon \underbrace{\sum_{n=1}^{N} |(\nabla_{\boldsymbol{r}}^2 + k^2) g(\boldsymbol{r}_n; \boldsymbol{\theta}_{\mathrm{NN}})|^2}$$

Loss for evaluating deviation from Helmhotlz eq



- **PINNs for reconstructing RIRs in time domain** [Pezzoli+ 2023]
  - RIRs measured by array of 100 mics are reconstructed using only 33 channels



Embedding physical properties in interpolation techniques

Four techniques to incorporate governing PDEs

- **Basis expansion into element solutions**
  - Plane wave expansion, spherical wave function expansion, equivalent source distribution
  - Expansion coefficients are obtained by linear regression
- **Kernel regression with constraint of governing PDEs**
  - Infinite dimensional extension of basis expansion
  - Kernel function to constrain the solution to satisfy Helmholtz eq
- **Feedforward NNs incorporating governing PDEs**
  - Feedforward NNs to estimate discrete target output
  - Setting output to expansion coefs or using approximate PDE loss
- **PINNs based on implicit neural representation**
  - NNs to implicitly represent continuous function
  - PDE loss computed by automatic differentiation

Outline

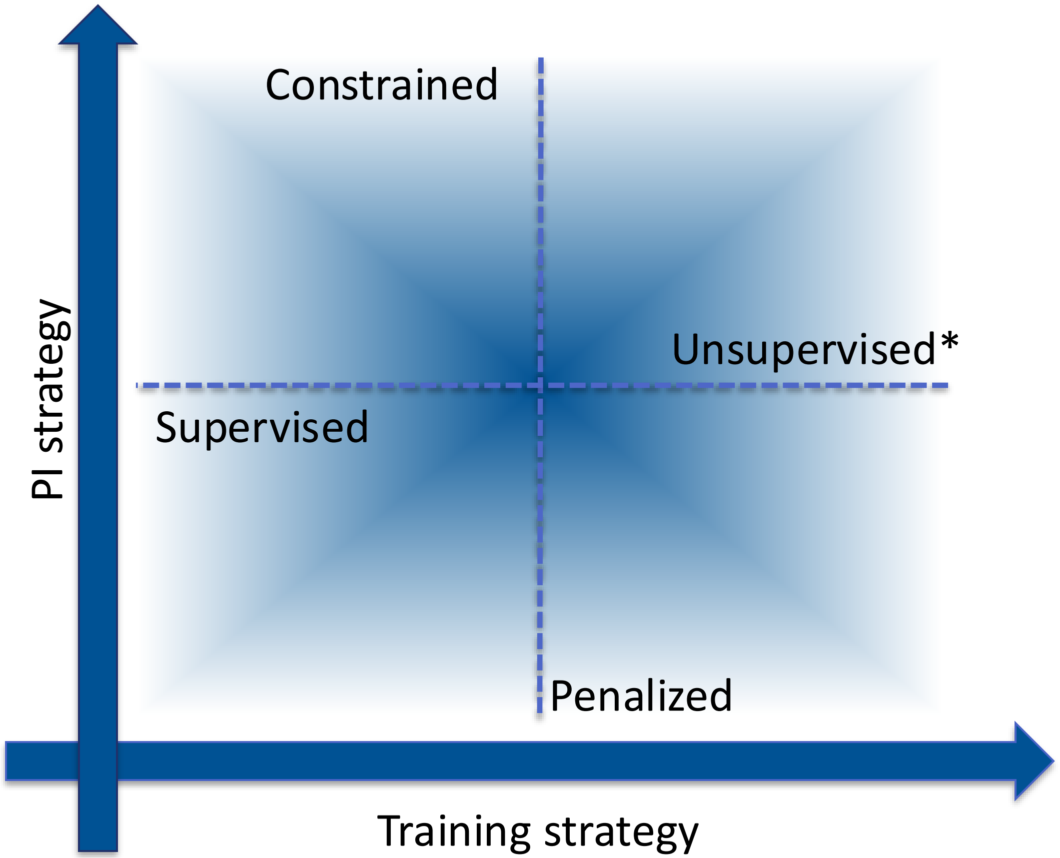
- What is sound field estimation
  - Problem setting
  - Applications
- Embedding physical properties in interpolation techniques
  - Basis expansion into element solutions
  - Kernel regression
  - Neural networks incorporating governing PDE
  - PINNs based on implicit neural representation
- **Current studies on sound field estimation based on PIML**
  - **Overview of state-of-the-art**
- **Outlook**
  - **Current limitations and future challenges**

overview

CURRENT STUDIES OF SOUND FIELD ESTIMATION BASED ON PIML

PIML techniques

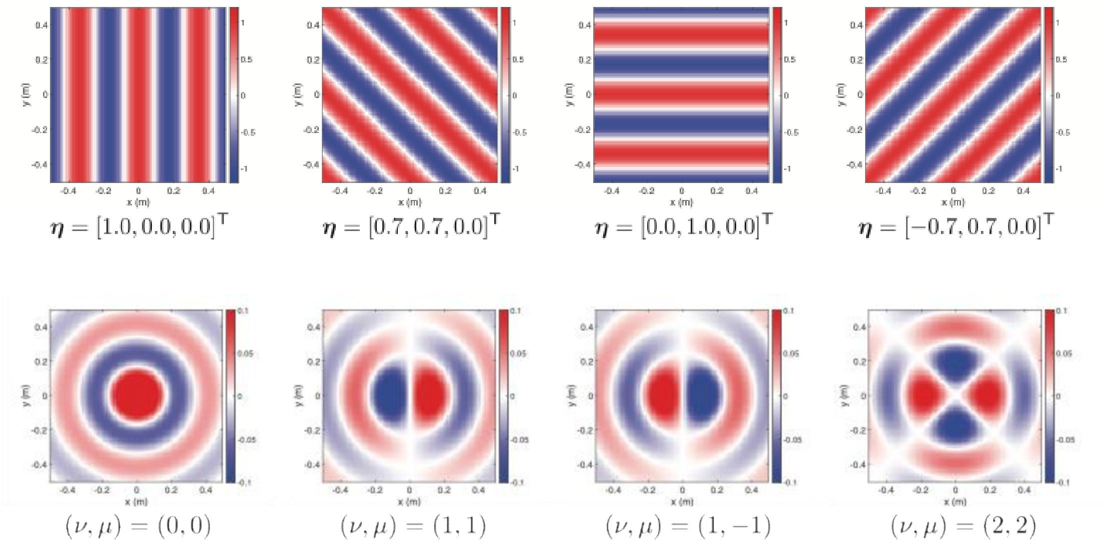
- Classification of current NN techniques based on
  - Training strategy
  - Strategy for adding physics priors

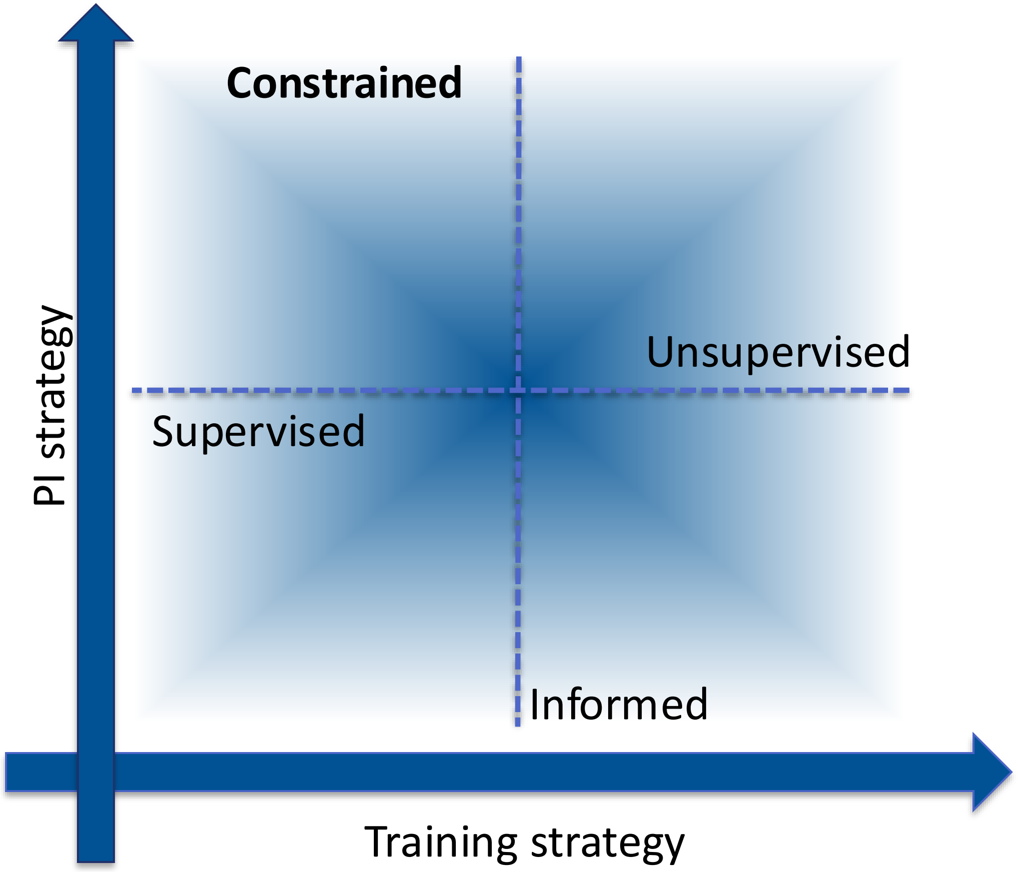


*we will focus on this class

Physics prior is introduced as:

- Constraint to adhere to physical model
  - Solutions of wave equation
- No deviations of the solution are allowed
  - Less flexibility in challenging scenarios

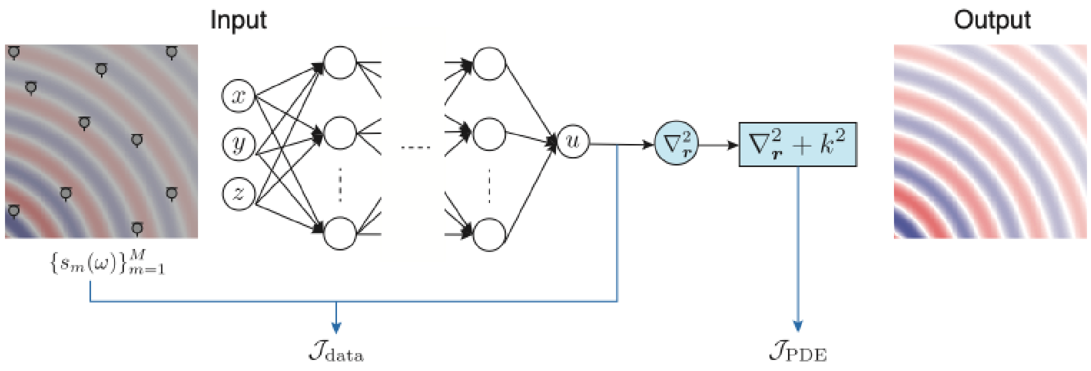


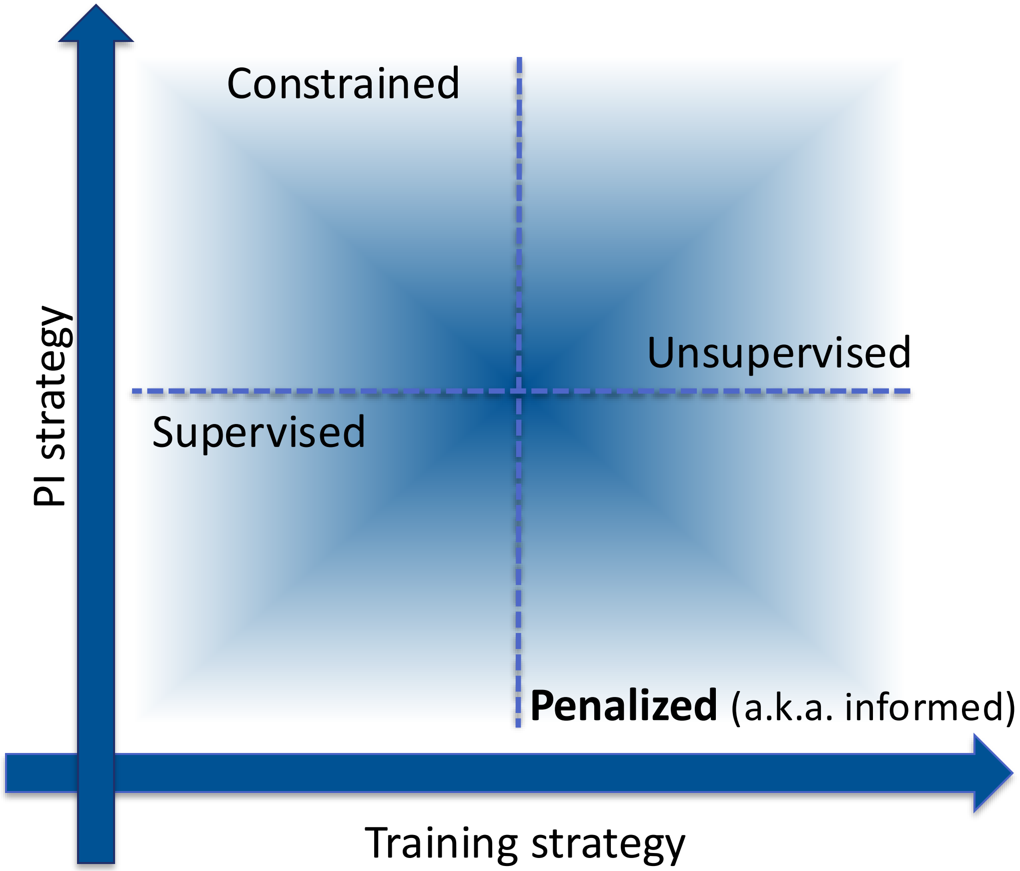


PIML techniques

Physics prior is introduced as:

- Penalization term of the optimization
  - Residual on wave/Helmholtz equation
- Small deviations of the solution are allowed
  - More flexibility in challenging scenarios



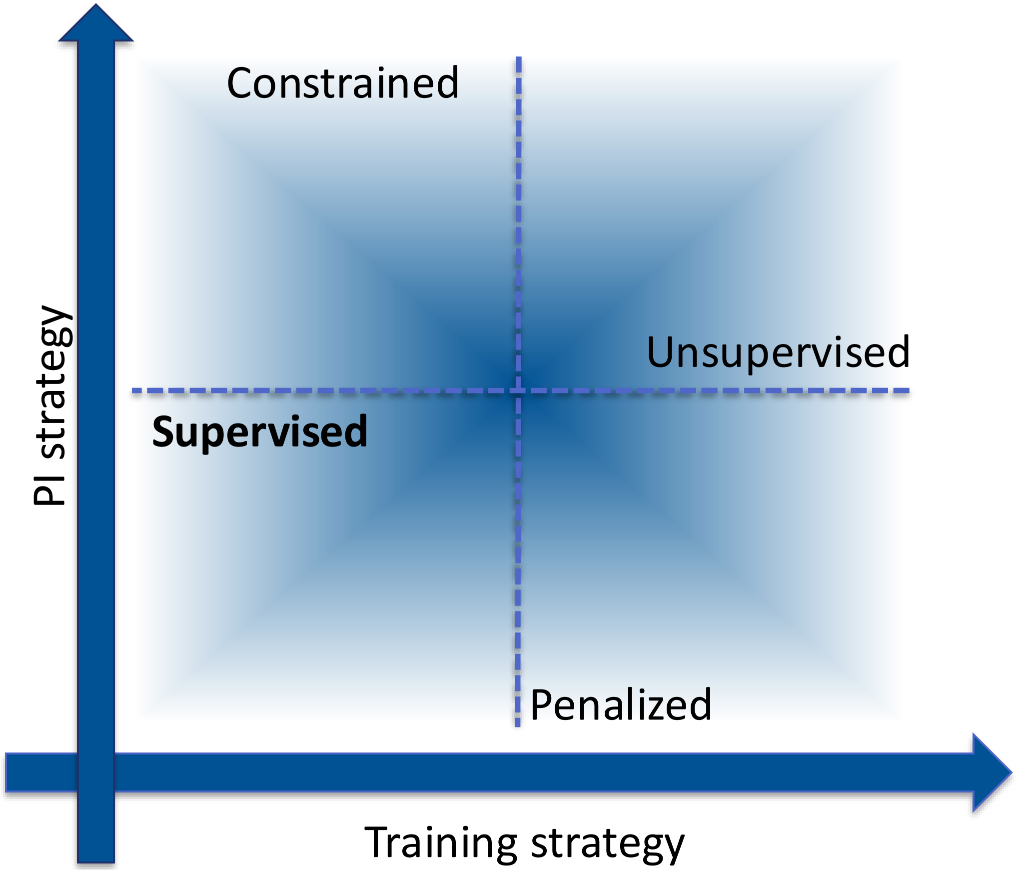


PIML techniques

PIML training approach is similar to standard ML

- Supervised: we have access to “ground truth”
  - Direct comparison between NN output and GT
  - Training dataset ≠ Test dataset
  - Common scenario for regression and classification

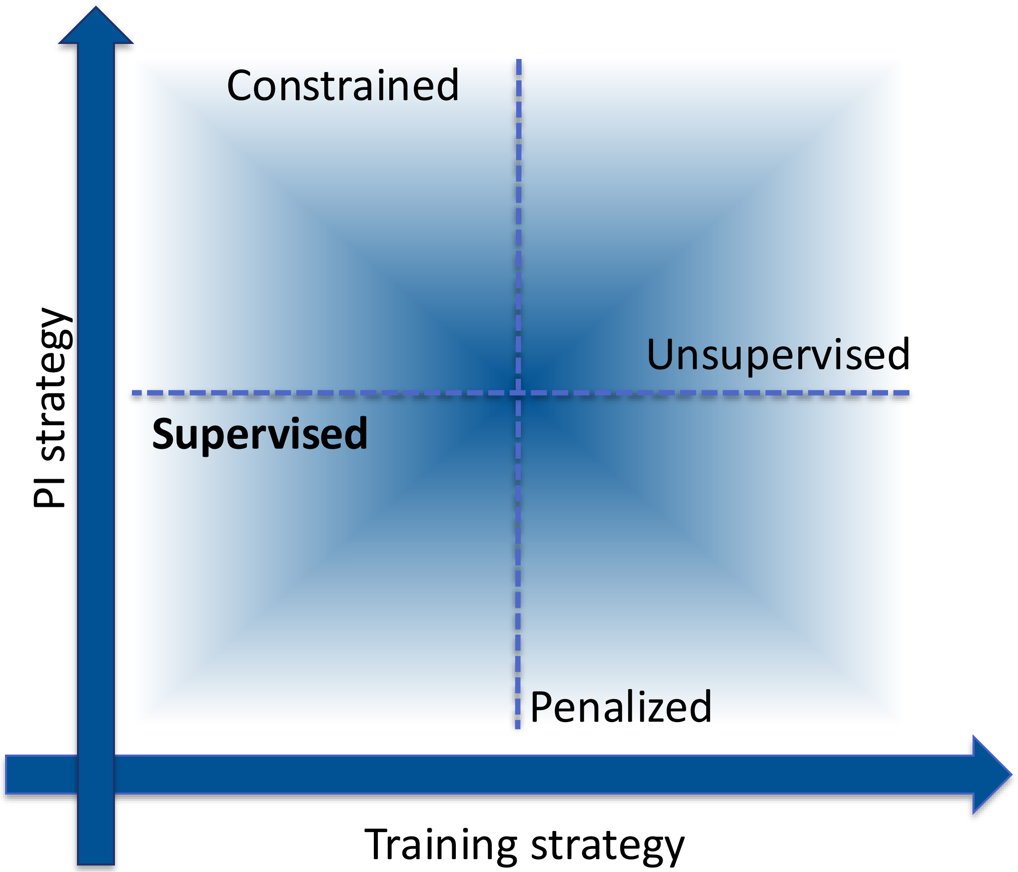
- Training stage
  - Dataset of measurements or simulations
- Test stage
  - Inference on new data



PIML techniques

PIML training approach is similar to standard ML

- Supervised: we have access to “ground truth”
  - Direct comparison between NN output and GT
  - Training dataset ≠ Test dataset
  - Common scenario for regression and classification



- Pros:
  - Exploit available data
  - Fast inference
- Cons:
  - Generalization is difficult

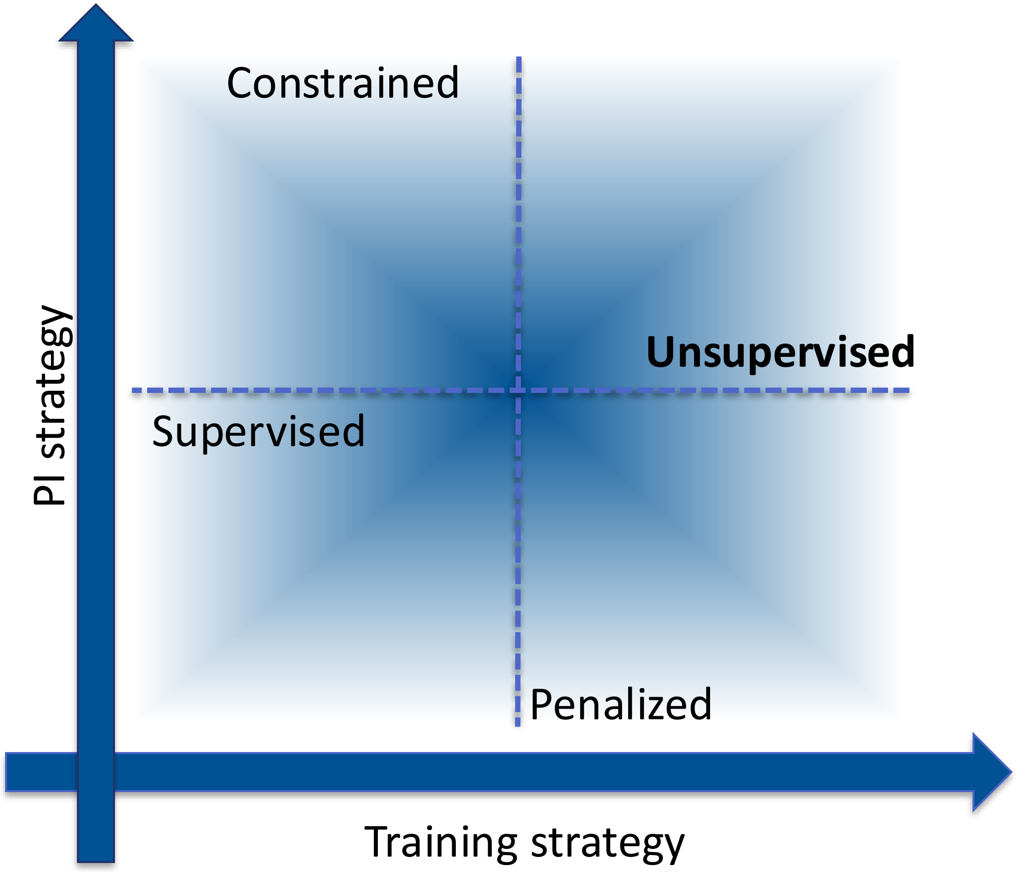
PIML techniques

PIML training approach is different from standard ML

- Standard meaning for unsupervised in ML:
  - No access to “ground truth”
  - Common scenario for clustering

- **Unsupervised**
  - no training dataset, “per-element” training
  - Same conditions for training and testing
  - ”overfit” the model

- Training stage
  - Only available measurements are used
- Test stage
  - Model applied on the same data

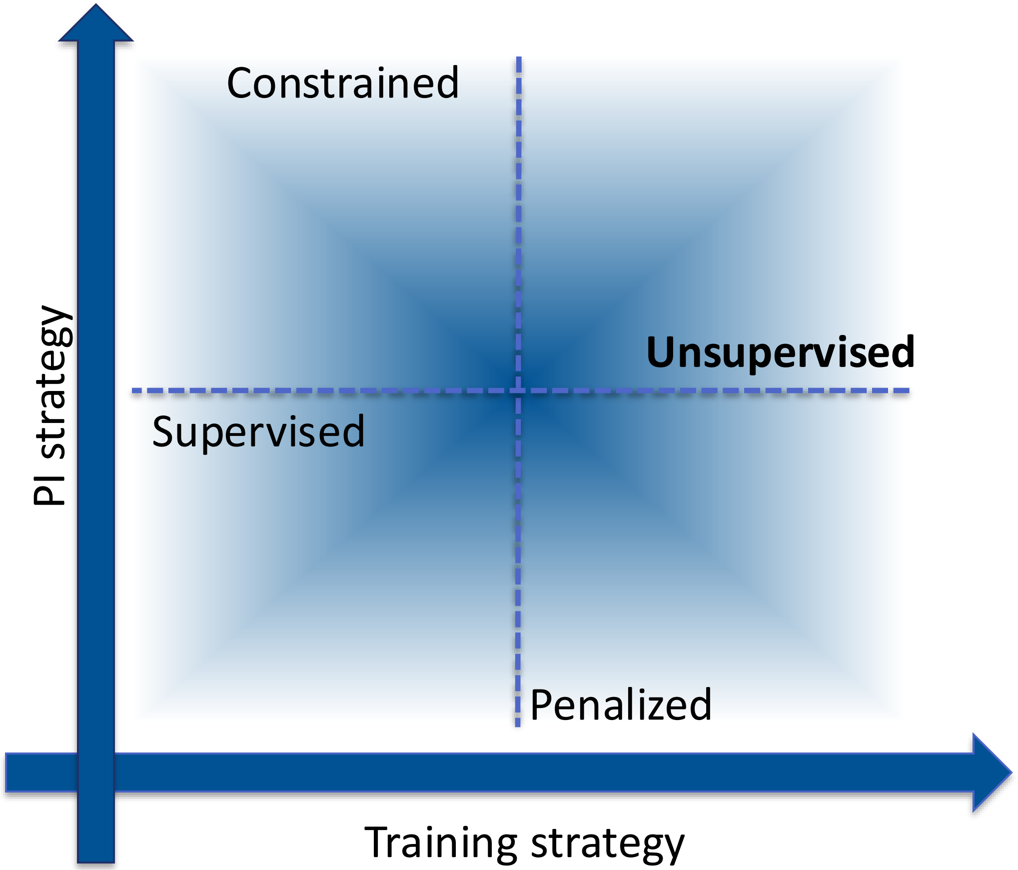


PIML techniques

PIML training approach is different from standard ML

- **Unsupervised**
  - no training dataset, “per-element” training
  - Same conditions for training and testing
  - ”overfit” the model

- Pros
  - No need for big training dataset
  - No “generalization” issues
- Cons
  - Does not exploit other available datasets
  - Needs re-training for new scenarios



PIML techniques

Paper	Supervised/Unsupervised	Estimator	Domain	Physical Property
Shigemi+ 2022	Supervised	Nonlinear	Frequency	Penalized
Karakonstatis+ 2023	Supervised	Linear	Frequency	Constrained
Olivieri+ 2024	Unsupervised	Nonlinear	Time	Penalized
Ribeiro+ 2024	Unsupervised	Linear	Frequency	Constrained

PIML techniques

Paper	Supervised/Unsupervised	Estimator	Domain	Physical Property
Shigemi+ 2022	Supervised	Nonlinear	Frequency	Penalized
Karakonstatis+ 2023	Supervised	Linear	Frequency	Constrained
Olivieri+ 2024	Unsupervised	Nonlinear	Time	Penalized
Ribeiro+ 2024	Unsupervised	Linear	Frequency	Constrained

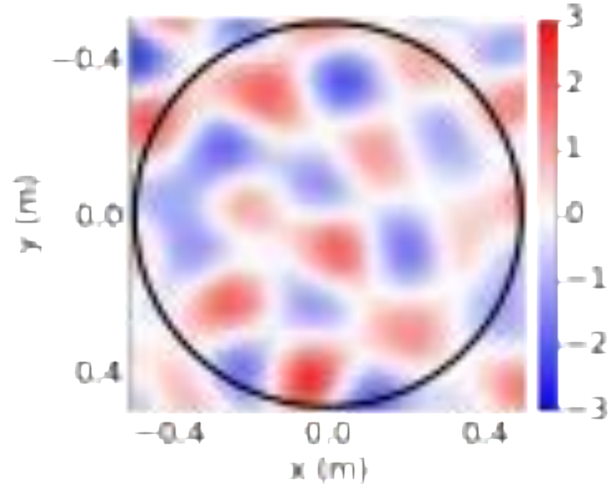
Ribeiro+ 2024

"Sound Field Estimation Based on Physics-Constrained Kernel Interpolation Adapted to Environment"

Juliano G. C. Ribeiro, Shoichi Koyama, Ryosuke Horiuchi, and Hiroshi Saruwatari

IEEE/ACM Trans. Audio, Speech, Lang. Process.

- KRR in combination with NN for carefully model the sound field
- Frequency-domain model, unsupervised, constrained



Neural kernel for sound field estimation

Reproducing kernel function adapted to acoustic environment using neural networks

- Kernel function with constrainst of Helmholtz eq is optimized to acoustic environment with the aid of neural networks [Ribeiro+ 2024]
  - Superposition of two kernel functions

$$\underbrace{\kappa = \kappa_{\mathrm{dir}}}_{\text{Directed kernel}} + \underbrace{\kappa_{\mathrm{res}}}_{\text{Residual kernel}}$$

  - **Directed kernel**: direct source and early reflections
  - **Residual kernel**: late reverberations and residual components

Neural kernel for sound field estimation

Reproducing kernel function adapted to acoustic environment using neural networks

➢ Directed kernel

– Directional weighting with weighted sum of (sparse) von Mises—Fisher distribution [Horiuchi+ 2021]

$$w_{\mathrm{dir}}(\boldsymbol{\eta};\boldsymbol{\beta},\boldsymbol{\gamma}) = \sum_{n=1}^{N} \gamma_n \frac{\mathrm{e}^{\beta_n \langle \boldsymbol{\eta}, \boldsymbol{d}_n \rangle}}{C(\beta_n)} \qquad (\|\boldsymbol{\gamma}\|_1 = 1)$$

➡ Sparsity constraint

$$\kappa_{\mathrm{dir}}(\boldsymbol{r}_1, \boldsymbol{r}_2) = \sum_{n=1}^{N} \gamma_n \frac{j_0\left(\sqrt{(\mathrm{j}\beta\boldsymbol{\eta} - k\boldsymbol{r}_{12})^{\mathsf{T}}(\mathrm{j}\beta\boldsymbol{\eta} - k\boldsymbol{r}_{12})}\right)}{C(\beta_n)}$$

Normalization constant

Neural kernel for sound field estimation

Reproducing kernel function adapted to acoustic environment using neural networks

Residual kernel

- Directional weighting with implicit neural representation

$$w_{\mathrm{res}}(\boldsymbol{\eta}; \boldsymbol{\theta}) = \mathrm{NN}(\boldsymbol{\eta}; \boldsymbol{\theta})$$: Implicit neural representation

$$\kappa_{\mathrm{res}}(\boldsymbol{r}_1, \boldsymbol{r}_2) = \int_{\mathbb{S}_2} w_{\mathrm{res}}(\boldsymbol{\eta}; \boldsymbol{\theta}) \mathrm{e}^{-\mathrm{j}k\langle \boldsymbol{\eta}, \boldsymbol{r} \rangle} \mathrm{d}\boldsymbol{\eta}$$

➡ Computed by numerical integration

Neural kernel for sound field estimation

Reproducing kernel function adapted to acoustic environment using neural networks

➢ Again, (positive-definite) kernel function is the sum of directed and residual kernels

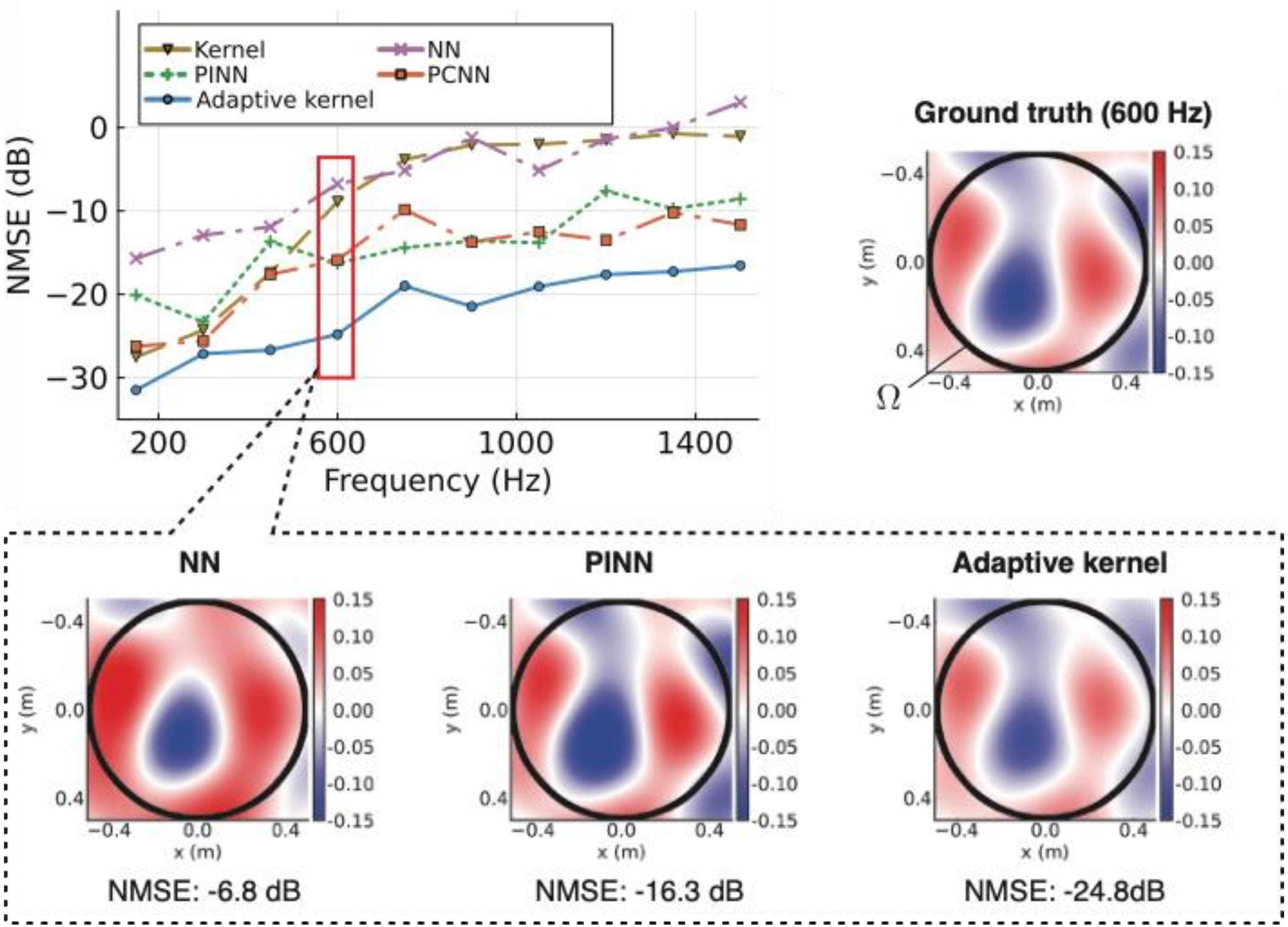
$$\kappa = \kappa_{\mathrm{dir}} + \kappa_{\mathrm{res}}$$

Directed kernel **Residual kernel**

– Hyperparameters $\boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\theta}$ are jointly optimized by a steepest-descent-based algorithm
– The method is physics-constrained
– Estimation process is still linear operation in freq domain based on kernel ridge regression

Neural kernel for sound field estimation

- Numerical experiment: T60: 400 ms, # mics: 41, spherical shell array



PIML techniques

Paper	Supervised/Unsupervised	Estimator	Domain	Physical Property
Shigemi+ 2022	Supervised	Nonlinear	Frequency	Penalized
Karakonstatis+ 2023	Supervised	Linear	Frequency	Constrained
Olivieri+ 2024	Unsupervised	Nonlinear	Time	Penalized
Ribeiro+ 2024	Unsupervised	Linear	Frequency	Constrained

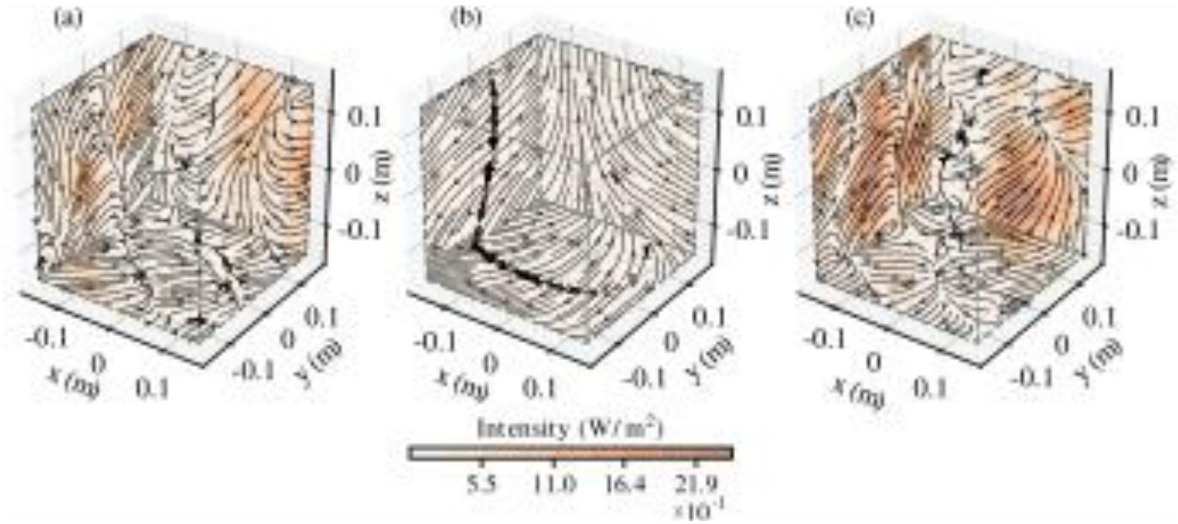
Olivieri+ 2024

“Physics-informed neural network for volumetric sound field reconstruction of speech signals”

Marco Olivieri, Xenofon Karakonstantis, Mirco Pezzoli, Fabio Antonacci, Augusto Sarti and Efren Fernandez-Grande

EURASIP J. Audio, Speech, Music Process.

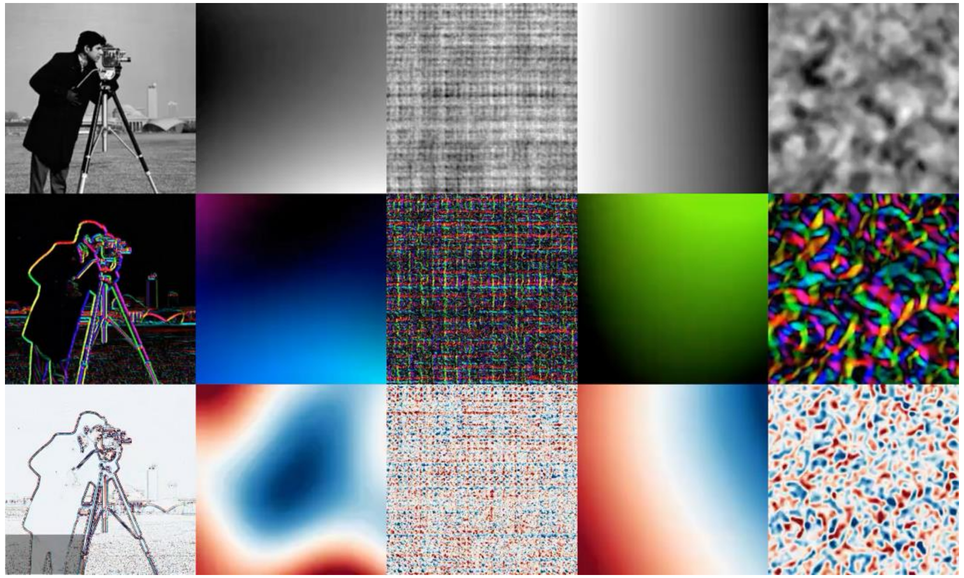
- Physics-informed neural network for sound field reconstruction
- Time domain model, unsupervised, penalized



PINN for sound field estimation

Implicit Neural Representation (INR)

- Implicit continuous differentiable representation of function f (a.k.a. Neural Field)
- Proved to be effective for different classes of signals (images, videos, point clouds etc.)

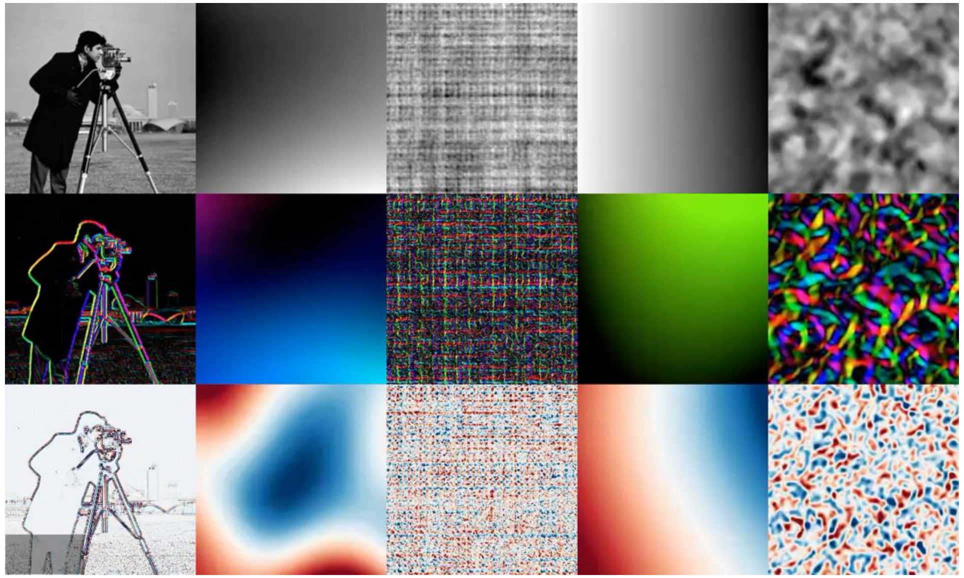


PINN for sound field estimation

Implicit Neural Representation (INR)

INR is used to implicitly represent the continuous function f

- Input is the domain $\boldsymbol{x}$ of f sampled in $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^{I}$
- Output are the value of f in $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^{I}$
- Typically, small MLPs are used



Sinusoidal representation networks (SIREN) [Sitzmann+ 2020]

- MLP structure with sinusoidal activations

Input

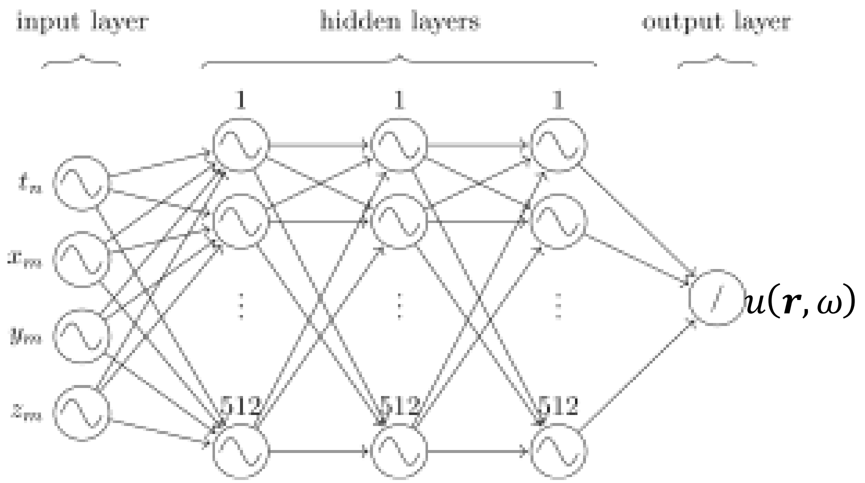
$$g(\boldsymbol{x};\boldsymbol{\theta}) = (\phi_L \circ \phi_{L-1} \circ \cdots \circ \phi_1)(\boldsymbol{x})$$

Learnable parameters

- Sine node

$$\phi_i(\boldsymbol{x}_i) = \sin\left(\omega_0 \boldsymbol{x}_i^T \boldsymbol{\theta}_i + \boldsymbol{b}_i\right)$$

Derivatives of SIREN are still SIREN



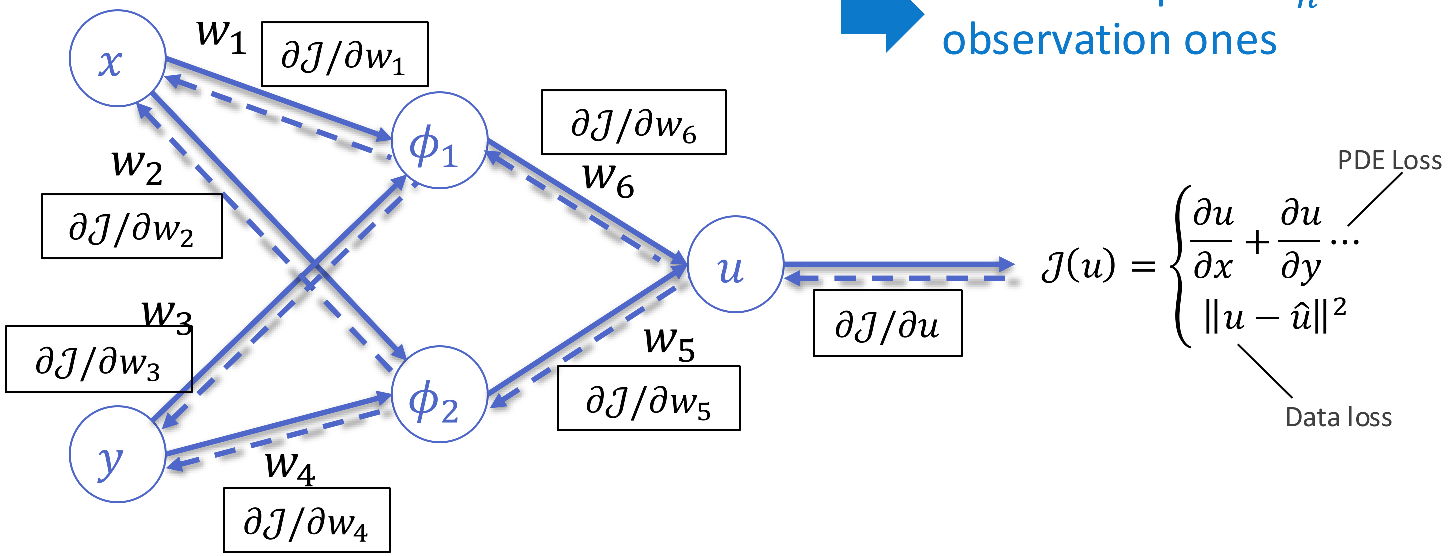
PINN for sound field estimation

Physics-informed SIREN (PI-SIREN)

- Being INR, SIREN allows for imposing constraints on its derivatives
  - Derivates are implemented using automatic differentiation
- Penalizing reconstruction using the residual of wave equation

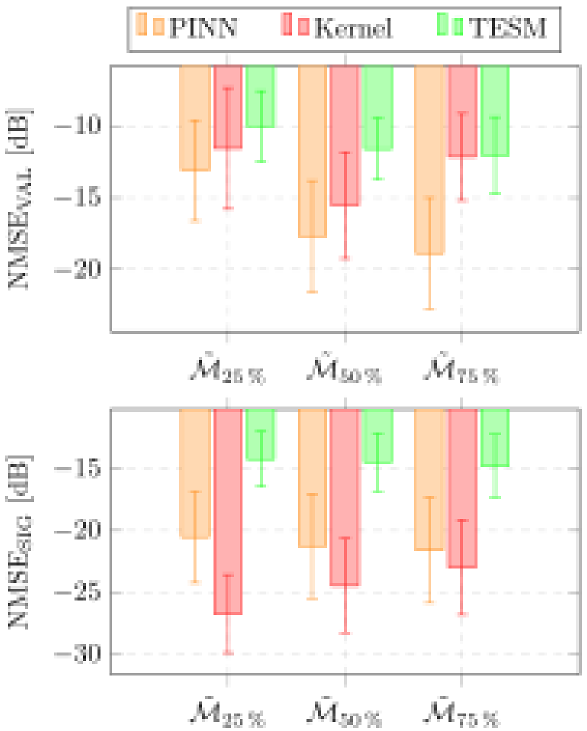
$$\mathcal{J}_{\mathrm{PDE}} = \sum_{n=1}^{N} \left| (\nabla_r^2 + \frac{1}{c^2}\frac{\partial^2}{\partial t^2}) g(\boldsymbol{r_n}, \boldsymbol{t}; \boldsymbol{\theta}_{NN}) \right|^2$$

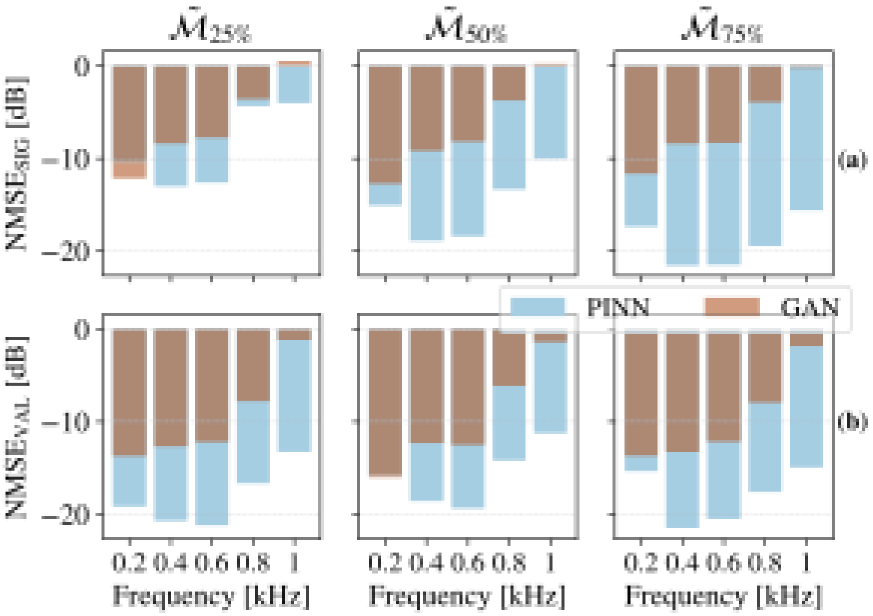
Evaluation point $\boldsymbol{r}_n$ can be different from the observation ones

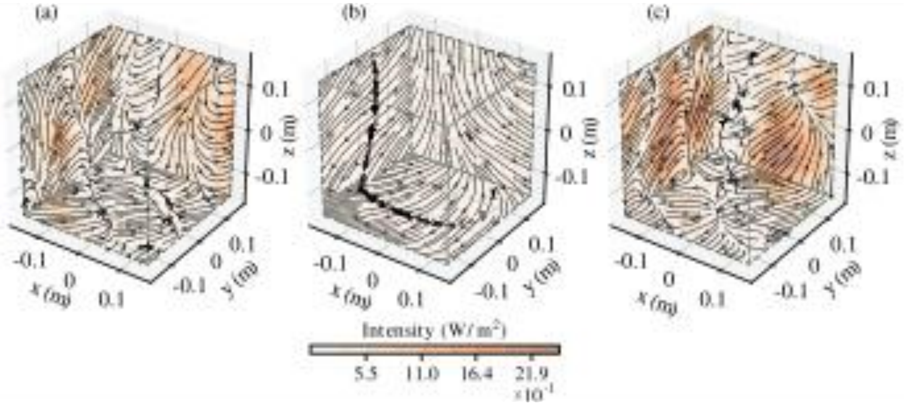


PINN for sound field estimation

- Evaluation on speech sound field using real measurements from MeshRIR [Koyama+ 2021]







[Olivieri+ 2024]

PINN for sound field estimation

PINNs are at the base of different sound field estimation works

- RIR Reconstruction using PI-SIREN
  - [Karakonstantis+ 2023, Pezzoli+ 2023, Karakonstantis+ 2024]

➡ First works using PI-SIREN

- Spherical microphones
  - [Chen+ 2023, Ma+ 2024]

➡ Variation in the architectures, frequency domain

- Nearfield acoustic holography
  - [Olivieri+ 2021]

➡ Physics loss with Kirchhoff-Helmholtz integral

- Sound field simulation
  - [Borrel-Jensen+ 2024]

➡ Solving forward problem using DeepONet overview

OUTLOOK

PIML Outlook

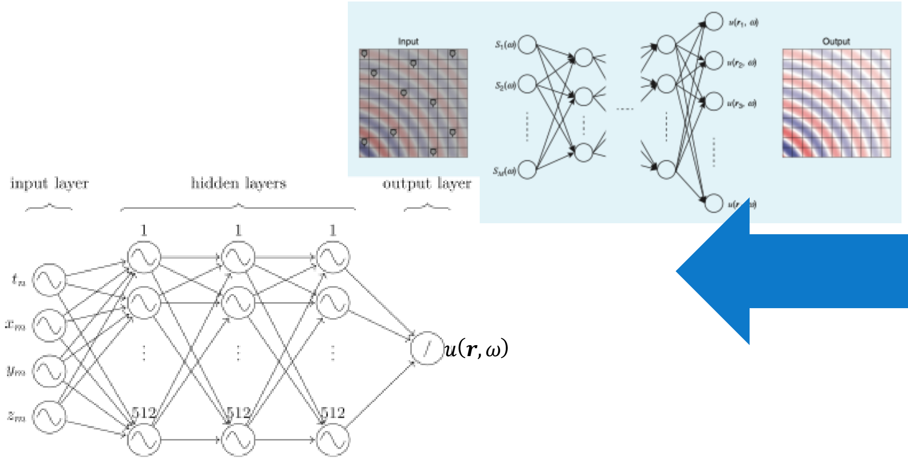
Sound field estimation took large advantage of PIML

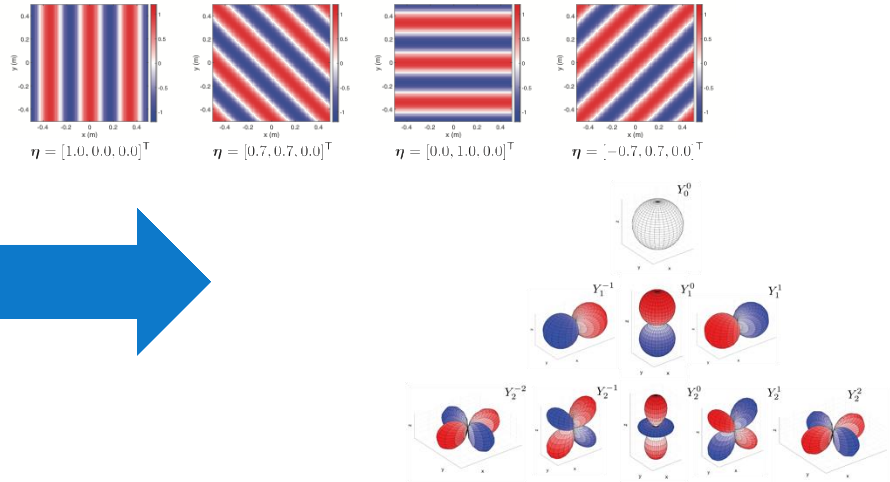
- Bridging ML with physical prior proved to be a winning approach

However, there are still major problems to be addressed

Machine learning

Domain knowledge





PIML Outlook

PIML sound field estimation: open challenges

We identified three main open challenges:

- Preparation of training data
- Mismatch between training and test data
- Neural network architecture design

PIML Outlook

PIML sound field estimation: open challenges

Preparation of training data

- Supervised methods potentially extract more information from data

However,

- High spatiotemporal resolution is required
- Large acoustic variations in different environments

➡ Simulations could be used but they have high computational cost

PIML sound field estimation: open challenges

Mismatch between training and test data

- Many parameters influence the acoustics
  - Source-receiver location
  - Environment geometry

➡ Cover extended ranges is unpractical

- Deviation between simulations and real world
  - Wave phenomena
  - Nonlinearities
  - Noise

PIML Outlook

PIML sound field estimation: open challenges

Neural network architecture design

- No clear methodology for architecture design
- Unsupervised methods mainly MLPs
  - Number of layers
  - Activations

  ➡ SIREN emerged as one of the main models
- Supervised methods
  - CNN
  - Generative methods
- Application dependent models

PIML Outlook

PIML sound field estimation: further observations

- Application of PIML for interior/mixed sound fields

  ➡ Some techniques could be applied here

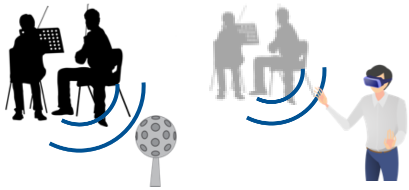
- Dependency on number and distribution of mics
  - Cover large areas with smallest number of microphones
  - Optimal placement is unclear

- Computational cost
  - Affects several applications e.g., noise control or HRTF interpolation
  - NN-based are mainly offline

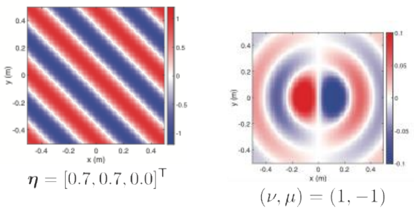
Conclusions

Physics-informed Machine Learning for Sound Field Estimation:

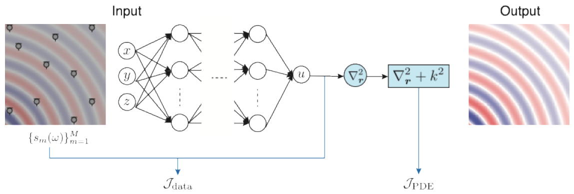
Fundamentals, state of the art and challenges



Problem: estimation of spatial sound



Solution: Inclusion of physics in machine learning methods



State of the art: methods and outlook