

弱形式型残差により支配方程式を学習する VPINN における動的重み付け法の検証

九州大学 学生会員 ○出口 翔大

九州大学 正会員 浅井 光輝

1. 緒言

機械学習手法により物理的な問題を評価するための方法論として、PINN (Physics-Informed Neural Network) が注目されている。PINN の特徴は、支配方程式の情報を学習に組み込むことで、多量のデータを必要とせずに学習を行える点にある。最近では、支配方程式の残差を選点法的に評価する PINN に対し、Petrov-Galerkin 法的に弱形式型の残差を評価する VPINN (Variational PINN) ⁽¹⁾へと発展している。VPINN は、FEM と同様に、解析対象となる領域を部分要素へと分割することで、各要素内で Petrov-Galerkin 法に基づき独立に残差を評価する学習が可能である。これにより、非線形性が強い例題などでは、誤差分布に応じて領域のサイズを調整するアダプティブな学習も可能となる。ただし、損失関数に対する重み付けには多様性があり、改良の余地が残されている。本研究では、著者らが提案した選点法型の PINN の動的重み付け法 (DN: Dynamic Normalization) ⁽²⁾を VPINN に適用し、その有効性を検証する。

2. 機械学習手法

2. 1. PINN: Physics-Informed Neural Network

まず、以下の Poisson 方程式を考える。

$$\nabla^2 u(\mathbf{x}) = f(\mathbf{x}) \quad \text{in } \Omega \quad (1)$$

$$u(\mathbf{x}) = g(\mathbf{x}) \quad \text{on } \partial\Omega \quad (2)$$

PINN はまず、問題に対する解 u をニューラルネットワークワーク $\hat{u}$ により近似する。入力 $\mathbf{x}$ 、出力 $\hat{\mathbf{y}}$ を持つ L 層のニューラルネットワークにおいて、第 l ($= 1, 2, \dots, L$) 層における順伝播 $\mathbf{z}^{(l)}$ は、以下の通りである ($\mathbf{x} \in \mathbb{R}^{f_{in}}$, $\hat{\mathbf{y}} \in \mathbb{R}^{f_{out}}$, $\mathbf{z}^{(l)} \in \mathbb{R}^{f_{hidden}^{(l)}}$)。

$$\mathbf{z}^{(l)} = \sigma^{(l)}(\mathbf{W}^{(l)}\mathbf{z}^{(l-1)} + \mathbf{b}^{(l)}) \quad (3)$$

ここで、 $\mathbf{z}^{(0)} = \mathbf{x}$ 、 $\mathbf{z}^{(L)} = \hat{\mathbf{y}}$ であり、 $\sigma^{(l)}(\cdot)$ は活性化関数と呼ばれる要素ごとに作用する非線形写像である (出力層でのみ恒等写像、i.e., $\sigma^{(L)}(\cdot) = \text{identity}(\cdot)$)。 $\mathbf{W}^{(l)}$ 、 $\mathbf{b}^{(l)}$ は第 l ($= 1, 2, \dots, L$) 層における重み、バイア

スであり、以下の損失関数の最小化により学習する。

$$\mathcal{L}_{\text{PINN}}(\boldsymbol{\theta}) = \lambda_{\text{PDE}}^{\text{Str}} \mathcal{L}_{\text{PDE}}^{\text{Str}} + \lambda_{\text{BC}} \mathcal{L}_{\text{BC}} \quad (4)$$

$$\mathcal{L}_{\text{PDE}}^{\text{Str}} = |\langle \nabla^2 \hat{u} - f, \delta \rangle|^2 \quad (5)$$

$$\mathcal{L}_{\text{BC}} = |\hat{u} - g|^2 \quad (6)$$

ここで δ は Dirac のデルタ関数、 $\mathcal{L}_{\text{PDE}}^{\text{Str}}$ は強形式 (Strong form) で記述された支配方程式との残差、 $\mathcal{L}_{\text{BC}}$ は境界条件との誤差である。

2. 2. VPINN: Variational PINN

VPINN⁽¹⁾ は元の方程式に対応した弱形式を学習に用いる。よって、損失関数を以下のように定義する。

$$\mathcal{L}_{\text{VPINN}}(\boldsymbol{\theta}) = \lambda_{\text{PDE}}^{\text{Var}} \mathcal{L}_{\text{PDE}}^{\text{Var}} + \lambda_{\text{BC}} \mathcal{L}_{\text{BC}} \quad (7)$$

$$\mathcal{L}_{\text{PDE}}^{\text{Var}} = |-\langle \nabla \hat{u}, \nabla v \rangle - \langle f, v \rangle|^2 \quad (8)$$

ここで、 v は試験関数であり、 $\mathcal{L}_{\text{PDE}}^{\text{Var}}$ は弱形式 (Variational / Weak form) で記述された支配方程式との残差である。

2. 3. DN: Dynamic Normalization

式(4)、(7)で各項に与える重み ($\lambda_{\text{PDE}}^{\text{Str}}$, $\lambda_{\text{PDE}}^{\text{Var}}$, λ_{BC}) は近似解 $\hat{u}$ の精度に関して大きく関与し、これをある特定の割合として与えることが多い⁽¹⁾。よって、著者らは重みの動的なチューニング法 (DN: Dynamic Normalization) により、近似解の高精度化が期待できることを示した⁽²⁾。DN では各損失項の勾配の ℓ^2 ノルムが同等になるよう、重みを以下のように定める。

$$\hat{\lambda}_{\text{BC}}^{(n)} = \frac{\|\nabla_{\boldsymbol{\theta}} \mathcal{L}_{\text{PDE}}^{\text{Str}}(\boldsymbol{\theta}^{(n)})\|_2}{\|\nabla_{\boldsymbol{\theta}} \mathcal{L}_{\text{BC}}(\boldsymbol{\theta}^{(n)})\|_2} \quad (9)$$

$$\lambda_{\text{BC}}^{(n)} = \beta \lambda_{\text{BC}}^{(n-1)} + (1 - \beta) \hat{\lambda}_{\text{BC}}^{(n)} \quad (10)$$

ここで、 $\|\cdot\|_2$ はベクトルの L^2 ノルム、 β ($\in [0, 1]$) は重みの動的な変化を平滑化するための指数減衰率であり、本研究では $\beta = 0.99$ とした。また、学習コストの増分を抑えるため、学習10エポックごとに動的重みを更新することとした。なお、各項の相対的な重要度を考慮するため、支配方程式に関する損失項に与える重みは1とした ($\lambda_{\text{PDE}} = 1$)。本研究では、式(9)の分母を $\nabla_{\boldsymbol{\theta}} \mathcal{L}_{\text{PDE}}^{\text{Str}} \rightarrow \nabla_{\boldsymbol{\theta}} \mathcal{L}_{\text{PDE}}^{\text{Var}}$ と置換して同様のアルゴリズムを適用することで、VPINN に対する DN の有用性を検証する。

キーワード 機械学習、深層学習、PINN (Physics-Informed Neural Network)、VPINN (Variational PINN)

連絡先 〒819-0395 福岡県福岡市西区元岡 744 番地

九州大学伊都キャンパス ウエスト2号館 11階 1102号室 構造解析学研究室 TEL092-802-3370

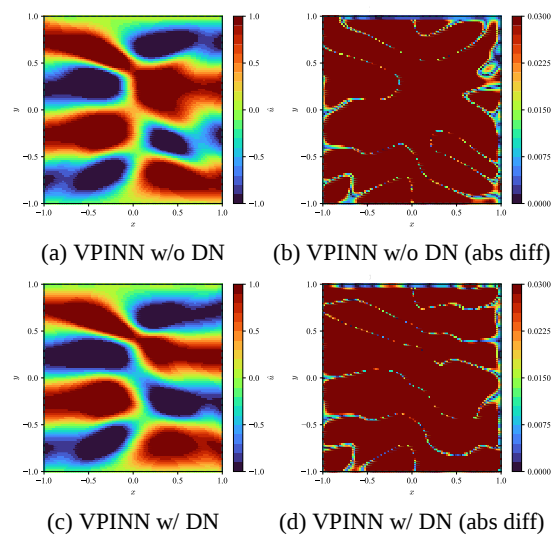


図-1 Poisson 方程式による検証（分割無し）

表-1 動的重みの有無による比較（分割無し）

Method	Rel. ℓ^2 err.	Mean sq. err.
w/o DN	5.72×10^{-1}	1.46×10^{-1}
w/ DN	6.22×10^{-1}	1.72×10^{-1}

3. 数値実験

動的重み付け法として DN を適用しない場合（w/o DN）と適用する場合（w/ DN）の結果を比較する．ここでは、既報⁽¹⁾で VPINN の検証に用いられた Poisson 方程式を考える．解を以下のように定め、右辺のソース項を用意して学習を行う．

$$u = (0.1 \sin(2\pi x) + \tanh(10x)) \sin(2\pi y) \quad (11)$$
本問題で用いるネットワーク構造は、 $L = 4$ 、 $f_{\text{hidden}} = 20$ 、 $\sigma(\cdot) = \tanh(\cdot)$ とした．また、学習には Adam を用い、50,000 エポックの学習を行った．

まず、1 要素飲みで学習を実施し、要素分割を行わない場合の結果を図-1、表-1 に示す．要素の細分割を行わない場合には、両者ともに適切な学習ができない．これは、先行研究⁽¹⁾と同様の結果であり、VPINN では、問題に応じて適切に要素分割することが重要である．

次に、各軸方向に向けて 4×4 の要素分割を行った場合の結果を図-2、表-2 に示す．図-2 より、領域を各要素に分割することで、動的重み付けの有無に関わらず、適切な学習ができている．また、表-2 より、動的重みを与えた方が近似精度は向上した．本問題では、10,000 エポック程度で動的重み $\lambda_{\text{BC}}^{(n)}$ が $\lambda_{\text{BC}}^{(n)} \cong 6.5$ 周辺に収束し、これより、境界条件の学習が加速された．図-2(b)より、従来型の VPINN では境界付近

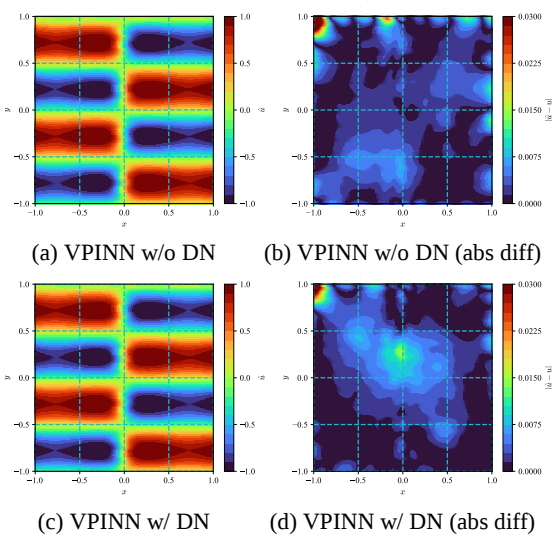


図-2 Poisson 方程式による検証（分割: 4×4 ）

表-2 動的重みの有無による比較（分割: 4×4 ）

Method	Rel. ℓ^2 err.	Mean sq. err.
w/o DN	6.55×10^{-3}	1.92×10^{-5}
w/ DN	5.97×10^{-3}	1.60×10^{-5}

での誤差が累積しているのに対し、動的重みの適用により、境界周辺での誤差が抑えられている（図-2(d)参照）．一方で、動的重み付けにより、領域内部、特に要素間の境界付近で大きな誤差が顕著に表れる結果となっている．

4. 結言

本研究では、VPINN に著者らが提案する動的重み付け法 DN を適用し、近似精度の向上を試みた．数値実験から、動的重みによる精度の向上を確認した．以上より、DN が VPINN にも拡張できる可能性を示した．ただし、まだ VPINN 特有の誤差発現の傾向が残されており、VPINN に固有の試験関数や求積法の選択方法と併せて追加検討が必要である．

参考文献

1) Kharazmi, E., Zhang, Z. and Karniadakis, G.E.M.: hp-VPINNs: Variational physics-informed neural networks with domain decomposition, *Computer Methods in Applied Mechanics and Engineering*, Vol. 374, pp. 113547, 2021.

2) Deguchi, S. and Asai, M.: Dynamic & norm-based weights to normalize imbalance in back-propagated gradients of physics-informed neural networks, *Journal of Physics Communications* (submitted).