

特集

ハイブリッドシミュレーション技術の動向

物理モデルと機械学習：
深層学習時代の展開

武石 直也

Physics Models and Machine Learning : Development in Deep Learning Era

1. はじめに

統計的機械学習は、タスクを解く方法をデータから直接獲得する手段として、研究開発を含む多くの分野で活発に研究・実践されている。特に科学の問題では現実世界の物理法則を無視した予測が起こらないよう注意が必要であり、物理法則を考慮することで予測の性能や解釈性を改善できるとも期待される。しかし多くの機械学習手法では必要な物理法則が組み込まれておらず、その学習のために多くのデータや計算が必要になる可能性がある。

本稿では、数理モデルの形で表される物理法則に関する知識（物理モデルとよぶ）を機械学習で活用する方法を紹介する。特に、データや損失関数を物理モデルから構築する方針と、機械学習モデルを物理モデルと直接組み合わせる方針のそれぞれについて、基本的な考え方と最近の展開を議論する。いずれの観点でも、特に深層学習を前提にした最近の展開を中心に紹介する。関連する応用例は数え切れないほど存在するため、不正確な分類を避けるという意味でもあまり具体的な例には立ち入らないことをご容赦いただきたい。関連するレビュー・サーベイ論文としては、文献13)、文献7)、文献25)、文献22)、文献27) などがある。特に化学工学では、Schweidtmannらによるハイブリッドモデルに関するレビュー論文¹⁶⁾が最近出版されている。本稿では関連話題をごく簡単に概観してキーワードを提供するにすぎないので、技術的な詳細はこれらの文献やその中で引用されている論文を参照いただきたい。



Naoya TAKEISHI
2018年 東京大学大学院工学系研究科航空宇宙工学専攻
現 在 東京大学大学院工学系研究科先端学際工学専攻 講師
連絡先；〒153-8904 東京都目黒区駒場4丁目6番1号
E-mail ntake@g.ecc.u-tokyo.ac.jp

2025年6月1日受理

2. 機械学習とドメイン知識

統計的機械学習は、訓練データについて計算される損失関数（例えば予測誤差）を小さくするように、仮説空間（モデルパラメータの空間）から要素（モデルパラメータ）を選ぶことで行われる。万能な予測モデルを有限のデータだけから獲得することは原理的に不可能であり、有用な予測のためには対象がもつべき性質の事前知識が必要である。「単純なモデルが好ましい」といった一般性の高い事前知識だけでなく、タスク特有のドメイン知識を適切に活用することで学習効率や予測性能を改善できると期待される。

物理モデルを機械学習で利用する方法は次のふたつに大別できる。まず、訓練データを物理モデルや対応するシミュレーションから生成したり、損失関数を物理モデルによって構成したりする方法があり、これは3節で紹介する。もうひとつは、機械学習モデルのなかに物理モデルを直接組み合わせるといったハイブリッドなモデリング方法であり、4節で紹介する。ふたつの方針の大きな違いは、推論時（テスト時）に物理モデルが直接参照されるかどうかである。訓練データや損失関数だけに物理モデルが反映されている場合は、それらが使われるのは基本的に訓練中のみであり、訓練後に物理モデルが常に使われる保証はない。一方、予測をなすモデルのなかに物理モデルが直接組み込まれていれば、訓練した後の予測でも常にそれが参照される。

ここでいう「物理モデル」は、場合によって異なるレイヤーの概念をさすことに注意されたい。物理現象の対称性（保存則）などの抽象度の高い概念から、微分方程式などによる具体的な数理モデルまで、様々な物理モデルを機械学習で活用し得る。本稿では、方程式やルール集合などの具体的な形で書かれていて、計算機での数値計算が構成できる数理的・経験的・統計的なモデルをおもな対象として考える。対称性に基づく機械学習も非常に活発な研究トピックではあるが、これについては例えば文献3) をご覧いただきたい。

3. 物理モデルに基づくデータ・目的関数

3.1 解作用素の学習

物理モデルの条件から解への写像（例えば、偏微分方程式の境界条件を入力として、ある時刻での解を出力とする写像）を学習することができれば、学習したモデルに新たな条件を入力して対応する解を高速に予測できる。ニューラル作用素⁸⁾とよばれるモデルはそのような目的に適したモデルとして知られている。特に、入出力をどちらも関数として扱うという点が特徴的である。つまり、入出力の各要素は常に決まった空間上の点で評価される必要はなく、自由な位置で自由な数だけ評価した関数の値を入出力とすることができる。解を予測するニューラル作用素を学習するためには、物理モデルを数値的に解いて生成したデータを用いる。つまり、入力となる条件をさまざまに変えながら物理モデルを解いて、それぞれに対する解を記録することで条件と解のペアの集合として訓練データを作る。

ニューラル作用素とよばれる一連のモデル⁸⁾のほかに、Transformerを利用した解作用素モデルが注目されている。例えば、文献2)では流れ場の形状（例えば自動車の形状）を点群として入力して、それをTransformerのようなself-attentionに基づくモデルで処理することで、形状表面上の圧力分布などを予測している。

3.2 逆問題解の学習

解作用素の場合とは逆に、物理モデルの解から条件への写像（例えば、微分方程式の解にあたる量の観測から微分方程式のパラメータを推定）の学習も活発に研究されている。パラメータから解を求めるという順問題に対して、解の観測からパラメータを求めるのは逆問題とよばれる。物理モデルによるシミュレーションを実行して逆問題を解く方法は総称してシミュレーションに基づく推論（simulation-based inference；SBI）や尤度なし推論（likelihood-free inference）とよばれる。これはもともと、パラメータの事後確率分布を推定するベイズ的逆問題の文脈で、シミュレーションの与える解の確率分布（つまりパラメータの尤度）の計算が難しい場合に、その計算をせず単にシミュレーションを実行して解を得るだけで推論したいという動機によるものである。

この問題を解く方法は機械学習だけではないが、ここでは機械学習ベースの方法に注目する。特に、深層生成モデルで観測からパラメータの確率分布を得る方法がさかんに研究されている。手法の概観は文献4)や文献28)を参照されたい。例えば、ニューラル事後分布推定（neural posterior estimation；NPE）⁵⁾とよばれる方法では、パラメータと観測のペアを事前分布とシミュレーションからそれぞれ生成し

て訓練データとし、観測に条件付けられたパラメータの分布（つまり事後分布）を正規化フローなどで学習する。最近では拡散モデル¹⁴⁾やフローマッチングモデル²⁴⁾によって事後分布を学習するという手法も提案されている。NPEは、入出力のペアがデータとして与えられるという意味で通常の教師あり学習と大差ない問題設定になっていることに注意されたい。つまり、事後分布推論という本来わりと注意が必要な問題を、より解きやすい問題に落とし込んでいると考えることもできる。

SBIに関するあるウェブサイト³⁰⁾では、SBIに関する論文情報がまとめられており、実に多様な分野で機械学習による逆問題解法が活用されていることがわかる。

3.3 深層学習による微分方程式解の表現

データを物理モデルから生成するのは別に、損失関数を物理モデルに基づいて定義することもできる。これらは実質的にはあまり違いがないようにも思われるが、問題設定次第でいずれかの見方が整理しやすいだろう。

微分方程式の解を深層ニューラルネットワークで表現して学習する方法として、physics-informed neural networks (PINN)¹²⁾やdeep Galerkin method (DGM)¹⁷⁾とよばれる方法がある。これらの方法では、位置や時刻を入力として解の値を出力とするニューラルネットワークを考える。これを微分方程式の残差や初期条件・境界条件との非合致度を損失関数として学習する。これにより、空間・時間の離散化によらず任意の位置・時間で評価できる解の表現を得られる。PINNの学習では、損失関数をどの位置・時刻で評価するかや、異なる位置・時刻についてどの順序で損失を小さくするか（いわゆるカリキュラム）など多くの設定が学習の成否に重要になる。文献23)ではこのようなテクニックについてまとめている。

どのような場合にPINNの学習に意義があるかは注意して考える必要がある。離散化によらない解表現が得られるとはいえ、多くの場合、時間的・空間的な補間によって任意の点での解を近似できる。また、数値計算にかかる時間と比べて、PINNの学習にかかる時間が短くなる問題設定は限られる。一方で、例えば解を位置・時間に関する微分可能な関数として得たい場合にはPINNが便利である可能性がある。

4. ハイブリッドモデリング

ここでは、3節で紹介したデータ・損失関数に物理モデルを組み込む方法とは対照的に、機械学習の仮説空間（モデル）の構成に物理モデルを直接利用する、つまり機械学習のモデルと物理モデルを直接つなげるというアプローチを紹介する。機械学習のモデルとして、特に深層ニューラルネットワークを想定する。

4.1 微分可能シミュレータ

深層ニューラルネットワークの学習は通常、損失関数の勾配に基づく最適化（確率的勾配降下法など）によって行われる。そのため、深層ニューラルネットワークに物理モデルを接続して学習するためには、物理モデルの微分も計算できる必要がある。例えば物理モデルが常微分方程式として表される場合、随伴法によって解の微分を計算できる。あるいは、PyTorch, TensorFlow, Jaxなどの自動微分機能を有するフレームワークで数値積分を実装してもよい。

微分の計算が可能な物理モデルの実装は「微分可能シミュレータ」とよばれる一分野を形成しており、多様な物理現象のシミュレータが微分可能な形で実装されている¹⁰⁾。なお、ここでいう「微分可能」とは数学的な微分可能性というよりも、微分、劣微分、またはそれらの近似を計算するための機能が備わっているという実装上の性質をさしていることに注意されたい。

「正しい」微分が常に学習にとって有利なわけではないという報告もある⁹⁾。例えば、カオティックな挙動を示すダイナミクスについての厳密な微分では、勾配が非常に大きな値になるため勾配法による最適化には不適切である。文献18)では、微分可能シミュレータによる厳密な報酬の勾配が強化学習では必ずしも有利ではなく、従来通りの有限差分に基づく近似勾配のほうが学習が安定する場合があることが報告されている。

4.2 デザインパターン

機械学習のモデルと物理モデルをつなげるといっても、様々なつなげ方が考えられる。例えば図1aのように、入力がかまず機械学習モデルで処理され、その結果が物理モデルのパラメータや初期条件等として利用されるというパターンがある。例えば、文献1)では感染症モデルのパラメータを人の移動データなどから機械学習で予測している。文献21)では、機械学習で予測した速度場をもとに移流方程式の時間発展で次ステップの気象状況を予測している。

逆に、図1bのように、入力がかまず物理モデルによって処理されてから機械学習モデルに渡されて予測するというパターンもある。これは、物理モデルによって機械学習モデルのための特徴抽出を行っているとも解釈できる。このパターンの特別な場合として、図1cのように、入力が物理モデルと機械学習モデルで並列に処理され、その何らかの結合として最終的な出力が得られるという構造もよく見られる。例えば、微分方程式のハイブリッドモデリングにおいて、もとの物理モデルによる項とニューラルネットワークによる項の足し算で微分方程式を構成することがよくある。例えば文献11)では、病気の進行を予測するために薬理学的な微分方程式にニューラルネットワークによ

る項を入れたモデルを使っている。文献29)では血糖応答の予測について同様のモデルが使われている。以上のようなデザインパターンについては、文献15)や文献16)で詳しく議論されている。

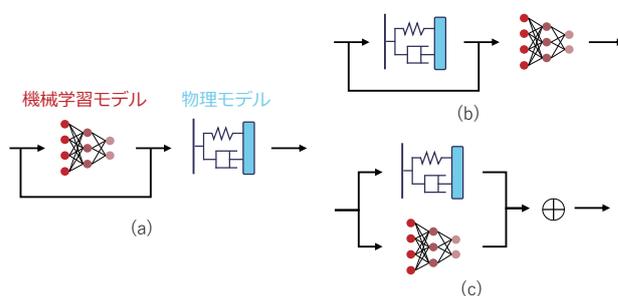


図1 ハイブリッドモデルのデザインパターン

4.3 学習方法

ハイブリッドモデルの形が図1bや図1cのパターンの場合には、学習の方法には注意が必要である。機械学習のモデルが深層ニューラルネットワークなどの表現力の高いものである場合には、物理モデル側の未知パラメータの値にかかわらずハイブリッドモデル全体はデータに適合してしまうからである。つまり、物理モデルで意味のないパラメータの値が選ばれていても、ほぼゼロの訓練誤差を達成できてしまう¹⁹⁾。表現力の高いモデルの過適合は通常の機械学習でも起こりうるが、ハイブリッドモデルでは物理モデルのパラメータの同定不可能性という形でも課題となる。

その解決策として、訓練誤差以外の学習の指針を導入することが考えられる。つまり、ハイブリッドモデルが満たすべき性質を、学習時の制約や正則化として導入する。正則化は過適合を抑える手段として一般的であるが、ハイブリッドモデルの正則化ではモデルの構造に応じた正則化が必要とされている。例えば、ハイブリッドな微分方程式を学習する際、ニューラルネットの出力のノルムを小さくするという正則化が使われる²⁶⁾。これは一般的なパラメータノルムの正則化とは異なることに注意されたい。また、ハイブリッドな生成モデルでは、物理モデルを補完するニューラルネットがない場合のモデルを仮想的に考え、そのモデルとハイブリッドモデルの差異を確率分布の非類似度として計算して小さくするという正則化が行われる²⁰⁾。

ハイブリッドモデルの正則化は、モデルの構造や問題の性質に応じて合理的に設計する必要がある。そのような設計を疑いなく行える事前知識や信念があれば問題ないが、現実的には探索的なデータ解析とおして正則化を手探りで設計・選択を行う必要がある。文献19)ではそのための方法を議論しているが、結局は分析者の信念によるところが大きく、ハイブリッドモデルの適切な学習についてはさらに議論が必要である。

4.4 今後の展望

ハイブリッドモデルは様々な分野で以前より活用されているが、深層学習の文脈でも手法の研究が進んでいる。筆者から見た今後の課題のひとつは、ハイブリッドモデルの構造探索である。現状ではほとんどの場合、物理モデルと機械学習モデルをどのようにつなげるかは専門家が設計する必要がある。しかし、多くのモデル構造を手で試行するのは現実的ではなく、また物理モデルに不要な部分が含まれる可能性がある場合には探索すべき構造はさらに多くなる。そこで、ハイブリッドモデルの構造を自動的に探索する手段が望ましい。深層ニューラルネットワークの構造探索の研究は以前から活発に行われており、この技術が活用できる可能性がある。また、大規模言語モデルによるコード生成やモデル評価が活用できる可能性がある⁶⁾。

ハイブリッドモデルの利点として、より良い予測性能が期待されることだけでなく、部分的な解釈性があげられることがよくある。つまり、人間が理解できるはずの物理モデルの部分はそのまま理解できるはずだ、ということだ。しかし、ここでいう理解は万能のものではないように思われる。例えば、どのような信念や知見に基づいて物理モデルのパラメータを同定したか適切に説明できなければ、物理モデルが何を担っているか議論できない。ハイブリッドモデルの解釈は、現象を理解するという根本的動機に従って慎重に行われる必要があるだろう。

5. おわりに

本稿では、物理モデルを機械学習に入れ込む方法として、訓練データや損失関数を物理モデルから作る方法と、機械学習のモデルに直接物理モデルをつなげるハイブリッドモデルを紹介した。筆者は、あくまで方法論としてはハイブリッドモデルに興味をもち研究している。モデル同士を直接つなげるという考え方が直感的に好まれる場合は多いが、実応用ではハイブリッドモデルは必ずしも意義と労力が見合う選択ではないことも少なくない。微分可能な物理モデルの実装を用意したり、物理モデルと機械学習モデルのつなげ方を注意深く設計したりするのと比べると、例えば物理モデルから訓練データを作って使うのは簡単に実装でき、またそれで十分な性能改善が得られることは珍し

くない。問題ごとの要求や得たい性質によってとるべき方法を見極めるのが重要になるだろう。

参考文献

- 1) Ank, Sercan Ö., Chun-Liang Li, Jinsung Yoon, Rajarishi Sinha, Arkady Epshteyn, Long T. Le, Vikas Menon, et al. : *Advances in Neural Information Processing Systems*, **33**, 18807-18(2020)
- 2) Bleeker, Maurits, Matthias Dorfer, Tobias Kronlachner, Reinhard Sonnlleitner, Benedikt Alkin, and Johannes Brandstetter. : arXiv:2502.09692(2025)
- 3) Bronstein, Michael M., Joan Bruna, Taco Cohen, and Petar Veličković. : arXiv:2104.13478, May 2(2021)
- 4) Cranmer, Kyle, Johann Brehmer, and Gilles Louppe. : *Proceedings of the National Academy of Sciences*, **117**, no.48, 30055-62(2020)
- 5) Greenberg, David S, Marcel Nonnenmacher, and Jakob H Macke. : *Proceedings of the 36th International Conference on Machine Learning*, 2404-14(2019)
- 6) Holt, Samuel, Tension Liu, and Mihaela van der Schaar. : *Advances in Neural Information Processing Systems*, **37**, 72170-218(2024)
- 7) Karniadakis, George Em, Ioannis G. Kevrekidis, Lu Lu, Paris Perdikaris, Sifan Wang, and Liu Yang. : *Nature Reviews Physics*, **3**, 422-40(2021)
- 8) Kovachki, Nikola, Zongyi Li, Burigede Liu, Kamyar Azizzadenesheli, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. : *Journal of Machine Learning Research*, **24**, no.89, 1-97(2023)
- 9) Metz, Luke, C. Daniel Freeman, Samuel S. Schoenholz, and Tal Kachman. : arXiv:2111.05803(2021)
- 10) Newbury, Rhys, Jack Collins, Kerry He, Jiahe Pan, Ingmar Posner, David Howard, and Akansel Cosgun. : *IEEE Access*, **12**, 97581-604(2024)
- 11) Qian, Zhaozhi, William R. Zame, Lucas M. Fleuren, Paul Elbers, and Mihaela van der Schaar. : *Advances in Neural Information Processing Systems*, **34**, 11364-83(2021)
- 12) Raissi, Maziar, Paris Perdikaris, and George E. Karniadakis. : *Journal of Computational Physics*, **378**, 686-707(2019)
- 13) Reichstein, Markus, Gustau Camps-Valls, Bjorn Stevens, Martin Jung, Joachim Denzler, Nuno Carvalhais, and Prabhat. : *Nature*, **566**, no.7743, 195-204(2019)
- 14) Rozet, François, and Gilles Louppe. : *Advances in Neural Information Processing Systems*, **36**, 40521-41(2023)
- 15) Rudolph, Maja, Stefan Kurz, and Barbara Rakitsch. : *Journal of Mathematics in Industry*, **14**, no.1, 3(2024)
- 16) Schweidtmann, Artur M., Dongda Zhang, and Moritz Von Stosch. : *Digital Chemical Engineering*, **10**, 100136(2024)
- 17) Sirignano, Justin, and Konstantinos Spiliopoulos. : *Journal of Computational Physics*, **375**, 1339-64(2018)
- 18) Suh, H. J. Terry, Max Simchowitz, Kaiqing Zhang, and Russ Tedrake. : *Proceedings of the 39th International Conference on Machine Learning*, 20668-96(2022)
- 19) Takeishi, Naoya, and Alexandros Kalousis. : *Proceedings of the 26th International Conference on Artificial Intelligence and Statistics*, 4089-4100(2023)
- 20) Takeishi, Naoya, and Alexandros Kalousis. : *Advances in Neural Information Processing Systems*, **34**, 14809-21(2021)
- 21) Verma, Yogesh, Markus Heinonen, and Vikas Garg. : *Proceedings of the 12th International Conference on Learning Representations*(2024)
- 22) von Rueden, Laura, Sebastian Mayer, Katharina Beckh, Bogdan Georgiev, Sven Giesselbach, Raoul Heese, Birgit Kirsch, et al. : *IEEE Transactions on Knowledge and Data Engineering*, **35**, no.1, 614-33(2023)
- 23) Wang, Sifan, Shyam Sankaran, Hanwen Wang, and Paris Perdikaris. : arXiv:2308.08468(2023)
- 24) Wildberger, Jonas, Maximilian Dax, Simon Buchholz, Stephen Green, Jakob H. Macke, and Bernhard Schölkopf. : *Advances in Neural Information Processing Systems*, **36**, 16837-64(2023)
- 25) Willard, Jared, Xiaowei Jia, Shaoming Xu, Michael Steinbach, and Vipin Kumar. : *ACM Computing Surveys*, **55**, no.4, 66 : 1-66 : 37(2022)
- 26) Yin, Yuan, Vincent Le Guen, Jérémie Dona, Emmanuel de Bézenac, Ibrahim Ayed, Nicolas Thome, and Patrick Gallinari. : *Proceedings of the 9th International Conference on Learning Representations*(2021)
- 27) Yu, Rose, and Rui Wang. : *Proceedings of the National Academy of Sciences*, **121**, no.27(2024) : e2311808121
- 28) Zammit-Mangion, Andrew, Matthew Sainsbury-Dale, and Raphaël Huser. : arXiv:2404.12484(2024)
- 29) Zou, Bob Junyi, Matthew E. Levine, Dessi P. Zaharieva, Ramesh Johari, and Emily Fox. : *Proceedings of the 41st International Conference on Machine Learning*, 62934-63(2024)
- 30) <https://simulation-based-inference.org/>, retrieved 2025-05-31