

ROLE OF PHYSICS IN PHYSICS-INFORMED MACHINE LEARNING

Abhishek Chandra,^{1,2} Joseph Bakarji,^{3,4} & Daniel M. Tartakovsky^{5,*}

¹Department of Electrical Engineering, Eindhoven University of Technology, The Netherlands

²Eindhoven Artificial Intelligence Systems Institute, Eindhoven University of Technology, The Netherlands

³Department of Mechanical Engineering, American University of Beirut, Beirut, Lebanon

⁴Artificial Intelligence, Computing and Data Science Hub, American University of Beirut, Beirut, Lebanon

⁵Department of Energy Science and Engineering, Stanford University, Stanford, California, USA

*Address all correspondence to: Daniel M. Tartakovsky, Department of Energy Science and Engineering, Stanford University, USA, E-mail: tartakovsky@stanford.edu

Original Manuscript Submitted: 3/11/2024; Final Draft Received: 5/2/2024

Physical systems are characterized by inherent symmetries, one of which is encapsulated in the units of their parameters and system states. These symmetries enable a lossless order-reduction, e.g., via dimensional analysis based on the Buckingham theorem. Despite the latter's benefits, machine learning (ML) strategies for the discovery of constitutive laws seldom subject experimental and/or numerical data to dimensional analysis. We demonstrate the potential of dimensional analysis to significantly enhance the interpretability and generalizability of ML-discovered secondary laws. Our numerical experiments with creeping fluid flow past solid ellipsoids show how dimensional analysis enables both deep neural networks and sparse regression to reproduce old results, e.g., Stokes law for a sphere, and generate new ones, e.g., an expression for an ellipsoid misaligned with the flow direction. Our results suggest the need to incorporate other physics-based symmetries and invariances into ML-based techniques for equation discovery.

KEY WORDS: symmetry preservation, dimensional analysis, equation discovery

1. INTRODUCTION

Advances in machine learning (ML) have impacted every field of science and engineering, including fluid mechanics [e.g., Brunton et al. (2020)]. In the disciplines that rely on scientific computing, ML has to cope with data scarcity rather than the deluge of “big data.” Recent work in this area includes surrogate modeling to accelerate computational fluid dynamics simulations (Kochkov et al., 2021), data-driven discovery of closure models (Bakarji and Tartakovsky, 2021;

Taghizadeh et al., 2020), and many variants of “physics-informed machine learning” (Karniadakis et al., 2021; Sharma et al., 2023).

Such efforts supplement scarce (experimental and/or computer-generated) data with general physical knowledge in order to train ML models. The role of physics in this process is to inform the formulation of a mathematical model, e.g., conservation laws underpinning the differential equations. This approach to scientific discovery is distinct from, and complementary to, empirical learning driven by experimentation which ignores the symmetry and invariance considerations inherent in physical systems (Wang et al., 2021). Dimensional relationship between the system states and system parameters is but one such manifestation.

A sphere of diameter D moving with velocity U in a fluid with density ρ and viscosity μ provides a classical illustration of this limitation. A standard ML strategy for discovering a relationship between drag force F_d and the parameters $\mathcal{S} = \{D, U, \rho, \mu\}$ involves the construction of a deep neural network (NN) $\mathcal{N}(\cdot)$ that takes $\mathcal{S}$ as its input and returns F_d as its output, i.e., learns a map $F_d = \mathcal{N}(\mathcal{S}; \mathbf{w})$. The weights $\mathbf{w}$ are computed by minimizing an average discrepancy, $\sum_i \|F_{d,i} - \mathcal{N}(\mathcal{S}_i; \mathbf{w})\|/N_{\text{exp}}$, between the observed and/or simulated data $\{F_{d,i}, \mathcal{S}_i : i = 1, \dots, N_{\text{exp}}\}$ from N_{exp} experiments for different parameter values $\mathcal{S}_i = \{D_i, U_i, \rho_i, \mu_i\}$. This procedure requires a large number of experiments (e.g., solves of Navier-Stokes equations), is likely to fit the observed data well, and is unlikely to generalize well to the conditions (parameter values) not used in the experiments (Chandra et al., 2023b; Kapoor et al., 2023a, 2024a; Kim et al., 2021). Our numerical experiments reported below demonstrate the shortcomings of this strategy.

Such learning is suboptimal, because it ignores a fundamental symmetry in the observed or simulated data that stems from their units. This symmetry manifests itself in the dimensional homogeneity of physically meaningful equations, i.e., in the requirement that every additive term in an equation must have the same units. (Other symmetries, e.g., the symmetry encapsulated in Noether’s theorem giving rise to conservation laws for Hamiltonian systems, are equally important but not considered here.)

It lays the foundation of dimensional analysis, i.e., the Buckingham Π theorem. For the problem above, the latter replaces the task of learning a four-dimensional map with the task of discovering its one-dimensional counterpart, $C_d = f(\text{Re})$, which maps Reynolds number $\text{Re} = DU\rho/\mu$ onto the properly rescaled drag force F_d that is known as drag coefficient C_d . One could deploy a neural network, sparse regression, or a polynomial to represent the univariate function $f(\cdot)$. Unlike data-driven techniques for construction or reduced-order models, most of which are based on singular-value decomposition [e.g., Lu and Tartakovsky (2020)], the order reduction resulting from the Π theorem comes without any loss of information (Yarin, 2012). This example demonstrates the role physics can and should play in physics-informed ML; it goes beyond using Navier-Stokes equations to generate data or constrain the loss function.

Empirical evidence suggests that dimensional analysis of physical data yields significant improvements in the performance of ML algorithms (Fukami and Taira, 2021; Jofre et al., 2020; Xie et al., 2022). Neural networks (Kapoor et al., 2023b,c, 2024b; Oppenheimer et al., 2023) and symbolic regression (Matchev et al., 2022) generalize better if their inputs and outputs are nondimensionalized. Based on the nondimensionalization of inputs, the concept of unit equivariance (Villar et al., 2023) imposes symmetries on ML algorithms, such as neural networks, ameliorating the problem of overfitting and, hence, improving their generalizability. The Buckingham theorem also informs ML-based approaches for discovery of dimensionless groups from observations data (Bakarji et al., 2022; Xie et al., 2022).

While our study does highlight the importance of dimensional analysis for improving ML-based strategies for equation discovery, its main goal is broader: It is to posit the general inadequacy of ML models that fail to account for fundamental symmetries of physical systems. Specifically, we demonstrate that the disregard for one such symmetry—physical invariance between dimensionless groups—results in failure of ML algorithms (neural networks and sparse regression) on the task of discovery of either equation-free models (in the form of a neural network) or analytical expressions (obtained via sparse regression) for the drag experienced by ellipsoidal objects moving through a viscous fluid.

This task is typical because most, e.g., fluid-mechanics, simulations of practical significance involve millions of degrees of freedom (state variables defined on large numerical grids) but a relatively small number of quantities of interest (QoI). For example, Navier-Stokes equations might be solved on a fine mesh in order to compute drag coefficient or lift. We refer to analytical expressions for such derived quantities as secondary laws, as in Stoke's law, the law of the wall, etc.

A recent trend in physics-informed ML is to express these derived laws and constitutive relations in terms of a deep NN, such as $\mathcal{N}(\mathcal{S})$ in the example above, a procedure that yields so-called equation-free models (e.g., Kapoor et al., 2022; Lennon et al., 2023; Souta et al., 2023, and references therein). An alternative strategy is to learn closed-form algebraic relations via, e.g., lasso-based sparse regression (e.g., Chandra et al., 2023a; Schmelzer et al., 2020). The former framework can handle a large number of input parameters, while the latter yields interpretable models. By reducing the number of model parameters, the Buckingham theorem enables these two frameworks to identify both equation-free models and analytical relations for drag coefficients.

2. SYMMETRY PRESERVING ML

When used to estimate the drag coefficient C_d of an object slowly moving in a viscous fluid, the Buckingham Π theorem yields a relation $C_d = f(\Pi_1, \dots, \Pi_N, \text{Re})$, where $f(\cdot)$ is an unknown function of Reynolds number Re and N unitless parameters $\mathbf{\Pi} = \{\Pi_1, \dots, \Pi_N\}$ characterizing the object's shape. A choice of these unitless groups is nonunique and subjective, requiring the application of human intelligence and prior understanding of the physical system. While various dimensionless variables may perform equally well in improving the performance of an ML algorithm, the crux lies in their physical significance, i.e., in the knowledge of physics. For example, the Reynolds number Re is a pivotal unitless group in fluid mechanics.

In addition to deep NNs, one could use multivariate Fourier series (Adcock, 2010) or orthogonal polynomials to approximate the function $f(\cdot)$ with a known function $\hat{f}(\cdot; \mathbf{w})$. In any case, the N_{par} parameters $\mathbf{w}$ are computed as a solution of the minimization problem,

$$\mathbf{w} = \underset{\mathbf{w}}{\text{argmin}} \frac{1}{N_{\text{exp}}} \sum_{i=1}^{N_{\text{exp}}} \|C_{d,i} - \hat{f}(\mathbf{\Pi}_i, \text{Re}_i; \mathbf{w})\|, \quad (1)$$

where $C_{d,i}$ is a value of the drag coefficient obtained in the i th experiment corresponding to the parameter values $\{\mathbf{\Pi}_i, \text{Re}_i\}$. This procedure imposes no penalty on the number of parameters, N_{par} ; the latter is specified instead by the neural-network architecture, a truncation rule for the Fourier series, or the polynomial degree. If the number of experiments, N_{exp} , is relatively small

or the number of fitting parameters, N_{par} , is relatively large—both common occurrences in ML-enabled discovery of secondary laws—then this procedure is likely to result in overfitting and poor generalizations.

We argue that arranging the data into unitless groups reduces the learning error by relying on physical knowledge to constrain the hypothesis class. That is because the total error is the sum of the estimation error and the approximation error, and a larger hypothesis class leads to better approximation but worse estimation. Constraining the hypothesis class to be physics-informed, e.g., decreasing dimensionality via the Buckingham theorem, improves generalization by biasing the model to honor the physics behind the data. In other words, an optimal fit of a learning algorithm (NN, sparse regression, etc.) to the unitless data would have a smaller test error than that of the equivalent algorithm trained on the dimensional data. Our goal is to demonstrate that to be true for NNs and sparse regression.

Sparse regression relies on physical knowledge to construct a library $\mathcal{M}(\Pi, \text{Re}) = \{\mathcal{M}_1(\Pi, \text{Re}), \dots, \mathcal{M}_{N_{\text{lib}}}(\Pi, \text{Re})\}$ of N_{lib} candidates for the derived law under consideration. (One such candidate, e.g., $\mathcal{M}_1 = 1/\text{Re}$, is provided by Stokes' law for the drag on a sphere). Next, a weighted sum of these candidates serves as an approximator of the drag law, $\hat{f}(\Pi, \text{Re}; \mathbf{a}) = \sum_k a_k \mathcal{M}_k$. The weights $\mathbf{a} = \{a_1, \dots, a_{N_{\text{lib}}}\}$ are determined via linear regression on the data from the N_{exp} experiments. We accomplish this via the least absolute shrinkage and selection operator (lasso) regression (Tibshirani, 1996),

$$\mathbf{a} = \underset{\alpha}{\operatorname{argmin}} \frac{1}{N_{\text{exp}}} \sum_{i=1}^{N_{\text{exp}}} \|C_{d,i} - \sum_{k=1}^{N_{\text{lib}}} \alpha_k \mathcal{M}_k(\Pi_i, \text{Re}_i)\|_2^2 + \beta \sum_{k=1}^{N_{\text{lib}}} \|\alpha_k\|_1, \quad (2)$$

where β is the regularization parameter. Typically, $N_{\text{lib}} \ll N_{\text{par}}$ to begin with, and the regularization term in Eq. (2) ensures that many of the regression coefficients a_i are zero. This strategy is designed to yield parsimonious and interpretable models.

We use the `lasso` subroutine from the open-source `scikit-learn` library to solve the minimization problem in Eq. (2). A value of the regularization parameter β is first chosen from a user-supplied set and its optimal value is then obtained using the `LassoCV` subroutine. The latter implements $\mathcal{K}$ -fold cross-validation to determine a suitable regularization strength β , dividing the data into $\mathcal{K}$ subsets, iteratively training on $\mathcal{K} - 1$ subsets, and validating on the remaining 1 fold. This procedure controls the number of coefficients α_n that are set to 0.

3. NUMERICAL EXPERIMENTS

We use black-box (implemented in Fluent) solutions of steady three-dimensional Stokes flow past a solid body to compute the drag coefficients $C_{d,i}$ for N_{exp} combinations of the unitless parameters $\{\Pi_i, \text{Re}_i\}$ with $i = 1, \dots, N_{\text{exp}}$. The body is placed at the center of a rectangular parallelepiped, whose size is sufficiently large for the boundaries not to affect the flow in the body's vicinity. The left and right sides of the parallelepiped serve as inlet and outlet on which unperturbed flow velocity $\mathbf{u} = (U, 0, 0)^\top$ is prescribed; the remaining four sides of the parallelepiped are impermeable to flow. No-slip boundary conditions are imposed on the body's surface.

Simulations are conducted for different values of U to generate data for $\text{Re} \in [0.001, 0.1]$, which is within the range for creeping flow. We use `LassoCV` to select an optimal value of the regularization parameter β from the set $\{0.0001, 0.001, 0.01, 0.1, 1, 10, 100\}$ and round the

model coefficients α_k ($k = 1, \dots, N_{\text{lib}}$), computed via Eq. (2), to three significant digits. Our experiments consider flows past a sphere (Section 3.1) and a prolate ellipsoid aligned (Section 3.2) and misaligned (Section 3.3) with the mean-flow direction. The first two scenarios admit analytical solutions for the drag coefficient C_d and serve to validate our approach to equation discovery. The third scenario is used to discover a new analytical expression for C_d as a function of Re , the ellipsoid aspect ratio, and the angle of attack.

3.1 Drag Coefficient for a Sphere

For creeping flow past a sphere of diameter D , the drag force F_d depends on free-stream velocity U and the fluid viscosity μ and density ρ . The Buckingham theorem reduces this dependence between five physical quantities to the functional relation, $\Pi = f(\text{Re})$, between the unitless group $\Pi = F_d/(\rho U^2 D^2)$ and the Reynolds number $\text{Re} = DU\rho/\mu$. Since the drag coefficient for any body is defined as $C_d = 2F_d/(\rho U^2 A)$, where A is the area of the body projected onto a plane perpendicular to the flow direction; and since, for a sphere, $A = \pi D^2/4$; the Buckingham theorem results in the relation $C_d = f(\text{Re})$. An analytical solution of Stokes equations, which is referred to as Stokes' law, identifies the function $f(\text{Re})$ to be $C_d = 24/\text{Re}$.

We use the lasso regression in Eq. (2) to “discover” the function $f(\text{Re})$ from $N_{\text{exp}}^{\text{tot}} = 20$ experiments corresponding to 15 values of Re uniformly covering the interval $[0.001, 0.01]$, and five values of Re uniformly covering the interval $[0.01, 0.1]$. These experiments result in data pairs $\{C_{d,i}, \text{Re}_i\}$ with $i = 1, \dots, N_{\text{exp}}^{\text{tot}}$, where each $C_{d,i}$ is obtained from a numerical solution of Stokes equations for $\text{Re} = \text{Re}_i$. The library of $N_{\text{lib}} = 5$ possible models, $\mathcal{M} = \{\mathcal{M}_1, \dots, \mathcal{M}_{N_{\text{lib}}}\}$, consists of positive and negative powers of Re , such that $\mathcal{M}_1 = \text{Re}^{-2}$, $\mathcal{M}_2 = \text{Re}^{-1}$, $\mathcal{M}_3 = \text{Re}^0$, $\mathcal{M}_4 = \text{Re}^1$, $\mathcal{M}_5 = \text{Re}^2$. This amounts to the expansion of $f(\text{Re})$ into a truncated Laurent-like series,

$$f(\text{Re}) \approx \hat{f}(\text{Re}) \equiv \sum_{k=1}^{N_{\text{lib}}} a_k \mathcal{M}_k(\text{Re}) = \sum_{n=-2}^2 \alpha_n \text{Re}^n.$$

The data set comprising $N_{\text{exp}}^{\text{tot}} = 20$ pairs $\{C_{d,i}, \text{Re}_i\}$ is subdivided into a training set of size $N_{\text{exp}} = 16$ and a testing set of size $N_{\text{exp}}^{\text{tot}} - N_{\text{exp}} = 4$. Cross-validation on the training set, carried out with the `LASSO`CV subroutine, yields the optimal sparsity parameter $\beta = 10^{-4}$. Using this value in Eq. (2) results in zero weights, $\alpha_n = 0$, for three out of the five candidates from the library and the fourth candidate becomes zero after rounding off to three significant digits. This procedure results in a learned model:

$$C_d = \begin{cases} 23.999/\text{Re} & \text{learned model,} \\ 24/\text{Re} & \text{Stokes' law.} \end{cases} \quad (3)$$

We use the relative percent error, $\mathcal{E} = |1 - C_{d,\text{dis}}/C_{d,\text{sim}}| \times 100\%$, to evaluate the prediction accuracy of this model, $C_{d,\text{dis}}$, with respect to either the exact solution (Stokes' law) or the numerical solution of the Stokes equations, both denoted by $C_{d,\text{sim}}$. The average error of $\mathcal{E}(\text{Re})$ on the interval $0.001 \leq \text{Re} \leq 0.1$ is $\mathcal{E} = 1.7 \times 10^{-7}\%$ and $\mathcal{E} = 3.5 \times 10^{-5}\%$ when the analytical and numerical solutions of Stokes equations are used as ground-truth, respectively. (The error in the latter case accounts for the numerical errors associated with solving Stokes' equations and computing the drag.)

To assess the robustness of this result, we rerun our algorithm on noisy training data. Each of the $N_{\text{exp}}^{\text{tot}} = 20$ values of the drag coefficient C_d is corrupted by adding a realization of zero-mean

Gaussian noise with standard deviation $\sigma_{\text{noise}} = 0.005, 0.01, 0.02, 0.05$, and 0.1 . Subsequently, the relationship between the noise-free Re and corrupted C_d is learned for five realizations of the noise for each value of σ_{noise} and the average of the five runs is recorded. The learned models range from $C_d = 23.957/\text{Re} + 2.631$ for $\sigma_{\text{noise}} = 0.005$ to $C_d = 23.121/\text{Re} + 0.0014/\text{Re}^2 + 58.122$ for $\sigma_{\text{noise}} = 0.1$. The relative deterioration in the learned model's quality is due to the sensitivity of sparse regression techniques to noise. While the lasso regression in Eq. (2) identifies the spurious terms in the learned models, for all noise levels the average error of $\mathcal{E}(\text{Re})$ does not exceed $10^{-3}\%$.

3.2 Drag Coefficient for a Prolate Ellipsoid Aligned with Flow

In the second set of experiments, we replace the sphere of diameter D with a spheroid that is elongated in the flow direction. This setting introduces an extra variable, as D is replaced with the lengths $2b$ and $2a$ along the spheroid's major and minor axes, respectively. The Buckingham theorem suggests that the ellipsoid's drag coefficient C_d is a function of both Reynolds number $\text{Re} = 2bU\rho/\mu$ and aspect ratio $\phi = b/a \geq 1$, leaving the functional form of $C_d = f(\text{Re}, \phi)$ unspecified. Happel and Brenner (1983) (pp. 148, 155) provide a closed form expression for $f(\cdot)$ by analytically solving Stokes equations.

We use the lasso regression in Eq. (2) to infer the function $C_d = f(\text{Re}, \phi)$ from $N_{\text{exp}}^{\text{tot}} = 100$ data triplets $\{C_{d,i}, \phi_i, \text{Re}_i\}$ with $i = 1, \dots, N_{\text{exp}}^{\text{tot}}$. These data are generated by computing C_d for 20 equidistant values of Re from the interval $[0.001, 0.1]$ and five discrete values of ϕ from the set $\{2, 3, 4, 5, 6\}$, i.e., for the 100 ordered pairs. This data set is subdivided into the training and testing data sets at the 4:1 ratio. The library used to discover the functional form of $f(\cdot)$ consists of algebraic powers $\text{Re}^n \phi^m$ for $-2 \leq n, m \leq 2$; i.e., it comprises $N_{\text{lib}} = 5 \times 5 = 25$ models. One can think of $\hat{f}(\text{Re}, \phi)$ as a bivariate Laurent-like series with parameters $\alpha \in \mathbb{R}^{N_{\text{lib}}}$. The `LassoCV` subroutine returns $\beta = 100$ and, after rounding off the obtained α_n to three significant digits, the lasso minimization procedure assigns 0 to all but two parameters, yielding

$$C_d = \frac{1}{\text{Re}} \begin{cases} 12.279/\phi + 12.076 & \text{learned model,} \\ \frac{32(\tau^2 - 1)^{-1/2}}{(\tau^2 + 1) \coth^{-1} \tau - \tau} & \text{H \& P model,} \end{cases} \quad (4)$$

where $\tau = (1 - \phi^{-2})^{-1/2}$ and "H & P" refers to the reference Happel and Brenner (1983).

The relative percent error of the learned model in Eq. (4), $\mathcal{E}(\text{Re}, \phi)$, with respect to the exact solution (Happel and Brenner, 1983) and numerical solution of Stokes equations is shown in Fig. 1. Regardless of the yardstick used for the comparison, the error does not exceed 0.05%, with its average over $(\phi, \text{Re}) \in [2, 6] \times [0.001, 0.1]$ of 0.0039% and 0.0146% when compared to exact or numerical data, respectively. In the limit $\phi \rightarrow 1$, i.e., when spheroid reduces to sphere, Eq. (4) yields $C_d = 24.355/\text{Re}$, which is within 1.5% of Stokes' law in Eq. (3). This result demonstrates the generalizability of the learned model to a parameter subspace ($1 \leq \phi < 2$) not explored during training.

An object in creeping flow experiences two forms of drag: skin friction drag (drag due to the friction between the object's surface and the flowing fluid) and form drag (drag due to the pressure difference between the front and back of the object). Stokes' law for a sphere lumps skin friction and form drag into a single term, as does the analytical solution Eq. (4) for oblate ellipsoids aligned with the flow. In contrast, the ML-derived expression in Eq. (4) explicitly

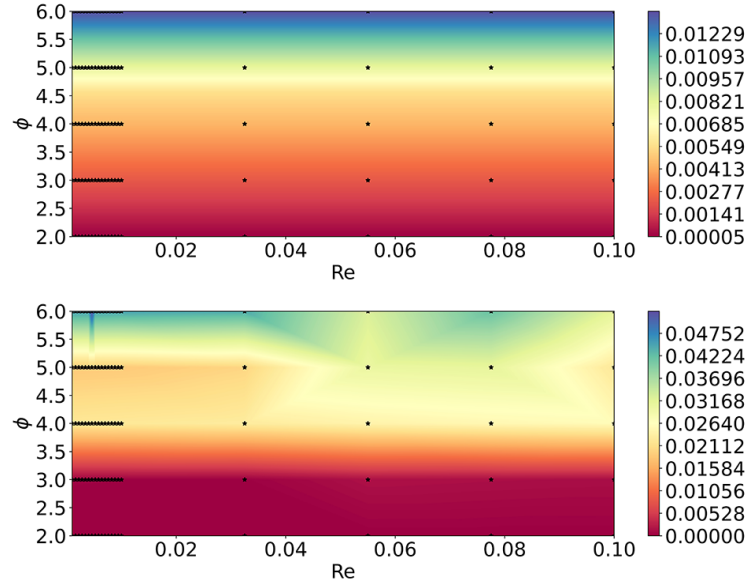


FIG. 1: Relative percent errors $\mathcal{E}(\text{Re}, \phi)$ of the discovered model of drag coefficient $C_d(\text{Re}, \phi)$ for creeping flow past a prolate ellipsoid aligned with the flow. The errors are with respect to either (top) the exact solution Happel and Brenner (1983) or (bottom) the numerical solution of the Stokes equations. The stars indicate data points in the parameter space.

accounts for form drag, $12.279/(\phi \text{Re})$, and skin friction drag, $12.076/\text{Re}$. This feature is an example of “explainable AI (artificial intelligence)” facilitated by the Π theorem.

3.3 Drag Coefficient for a Misaligned Prolate Ellipsoid

In the third set of experiments, we allow the spheroid to be misaligned with the flow, such that its major axes form an acute angle $\theta \in [0, \pi/2]$ with the flow direction. The Buckingham theorem now yields a functional relationship with an extra degree of freedom, $C_d = f(\text{Re}, \phi, \theta)$. In place of an exact analytical expression for $f(\cdot)$, one has to resort to empirical/heuristic relations, of which there are many (e.g., Andersson and Jiang, 2019; Hölzer and Sommerfeld, 2008; Livi et al., 2022; Ouchene et al., 2016).

We use the lasso regression in Eq. (2) to infer the function $C_d = f(\text{Re}, \phi, \theta)$ from $N_{\text{exp}}^{\text{tot}} = 500$ data quadruplets $\{C_{d,i}, \phi_i, \theta_i, \text{Re}_i\}$ with $i = 1, \dots, N_{\text{exp}}^{\text{tot}}$. The data are generated by computing C_d for the values of Re and ϕ from Section 3.2 and five discrete values of θ from the set $\{0, \pi/8, \pi/4, 3\pi/8, \pi/2\}$, i.e., for the 500 ordered triplets. This data set is subdivided into the training and testing data sets at the 4:1 ratio, such that $N_{\text{exp}} = 400$. The library used to discover a functional form of $f(\text{Re}, \phi, \theta)$ is the concatenation of two libraries. The first is the library used for the prolate ellipsoid aligned with the flow; it contains $N_1 = 25$ models. The second comprises algebraic powers $\text{Re}^n (\phi - 1)^m \theta^q$ with the integer exponents $(n, m, q) \in [1, 2] \times [0, 2] \times [-2, 2]$; i.e., it contains $N_2 = 2 \times 3 \times 5 = 30$ models. The total number of candidates in our library is $N_{\text{lib}} = N_1 + N_2 = 55$. The `LassoCV` subroutine yields $\beta = 100$, and identifies all but three of 55 weights to be 0 after rounding off, so that

$$C_d = \frac{1}{\text{Re}} \left[\frac{14.383}{\phi} + 11.443 + (\phi - 1)(0.784\theta + 0.038\theta^2) \right]. \quad (5)$$

The relative percent error of the learned model, $\mathcal{E}(\text{Re}, \phi, \theta)$, is plotted in Fig. 2 for $\text{Re} = 0.001$. The yardstick here is a numerical solution of Stokes equations. The errors are comparable with those from the second experiment (Fig. 1), with the average value of 0.127% over the hypercube $(\phi, \theta) \in [2, 6] \times [0, \pi/2]$. For $\theta = 0$, the spheroid is aligned with the flow, and the learned model in Eq. (5) reduces, approximately, to its counterpart in Eq. (4). For $\phi = 1$, the spheroid becomes a sphere, the θ -term vanishes (as it should since a sphere has no orientation), and Eq. (5) yields $C_d = 25.273/\text{Re}$, which is within 5.5% of Stokes' law, Eq. (3). Since our model Eq. (5) has been trained on the data for $2 \leq \phi \leq 6$, this result demonstrates the model's consistency and generalizability.

The drag coefficient C_d in Eq. (5) has three terms: the first two represent skin friction and form drag, as discussed in the previous subsection; the third term represents the impact of the orientation angle θ . It contributes mostly to the form drag, as it depends on the shape and size of the submerged body. The proposed approach provides a clear distinction between the different types of drag, a feature lacking in the traditional analytical formulae.

Table 1 collates values of C_d , and corresponding prediction errors $\mathcal{E}$, for $\text{Re} = 0.1$, $\phi = 6$, and three values of θ . These values are alternatively computed with Eq. (5) and with the empirical correlations from Hölzer and Sommerfeld (2008), Ouchene et al. (2016), Andersson and Jiang (2019), and Livi et al. (2022). Our parsimonious model has the best *overall* performance in approximating the drag coefficient over the full range of the angle of attack θ .

3.4 Lasso Regression without Dimensional Analysis

To isolate the benefit of dimensional analysis for ML-enabled discovery of secondary laws, we compare the results reported for a sphere with those obtained via the lasso regression on unprocessed data $\{F_{d,i}, U_i, a_i, \rho_i, \mu_i\}$ for $i = 1, \dots, N_{\text{exp}}$ from the same $N_{\text{exp}} = 20$ numerical experiments. The values of the unprocessed data are chosen so that their combination results in the same Re values used for all the experiments. We consider a library with $N_{\text{lib}} = 625$ candidates and corresponding weights α , which are represented by a six-variate truncated Laurent-like series,

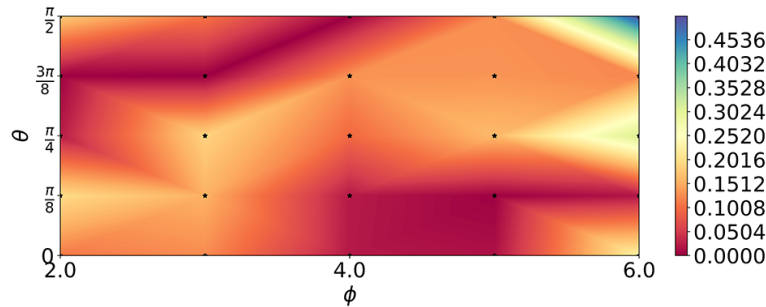


FIG. 2: Relative error $\mathcal{E}(\text{Re} = 10^{-3}, \phi, \theta)$ of the model of drag coefficient $C_d(\text{Re}, \phi, \theta)$ for a prolate ellipsoid misaligned with the flow direction for simulated data. The stars indicate data points in the parameter space.

TABLE 1: Drag coefficient, C_d , for creeping flow past an inclined prolate ellipsoid. It is alternatively estimated with our model and with four empirical relations—ER1 (Ouchene et al., 2016), ER2 (Hölzer and Sommerfeld, 2008), and ER3 (Andersson and Jiang, 2019)—for $Re = 0.1$, $\phi = 6$, and three angles of attack θ

θ	Learned		ER1		ER2		ER3	
	C_d	$\mathcal{E}$	C_d	$\mathcal{E}$	C_d	$\mathcal{E}$	C_d	$\mathcal{E}$
0	144.75	4.30%	139.58	0.58%	151.91	9.46%	138.39	0.27%
$\pi/4$	164.38	3.88%	166.40	2.70%	178.79	4.54%	174.22	1.87%
$\pi/2$	184.02	5.88%	175.90	10.03%	205.68	5.19%	203.44	4.05%

$$\hat{F}_d = \sum_{k_1=-2}^2 \cdots \sum_{k_4=-2}^2 \alpha_{k_1 k_2 k_3 k_4} U^{k_1} a^{k_2} \rho^{k_3} \mu^{k_4}. \quad (6)$$

The use of `LassoCV` to solve the minimization problem in Eq. (6) now yields a model for the drag force F_d that contains nine terms, identifying the remaining 616 weights as 0.

The prediction accuracy of this model accuracy is comparable to that of the model in Eq. (3). Yet, it fails to preserve the system's symmetry that is encapsulated in its invariance to the physical variables. Although five different combinations of $\{U, a, \rho, \mu\}$ leading to the same Re should yield the same value of drag force F_d , the learned model yields disparate predictions of the drag-coefficient C_d (Fig. 3) and completely misses the trend of the relationship between the physical variables. All five relationships learned without recourse to dimensional analysis are significantly different and provide incorrect predictions of the drag coefficient. Additionally, the nine-term model discovered without dimensional analysis is unwieldy and largely uninterpretable. These findings demonstrate the importance of dimensional analysis in ML-enabled discovery of secondary laws.

It is possible that a better sparsifier (e.g., the deployment of the L_0 -norm or sequential feature selection) or a more sophisticated approach that reduces noise before fitting might identify a better model without the data nondimensionalization. However, the trial and error, computational cost, and effort required to properly tune the hyperparameters for such a task amplify the reduction in estimation error gained through the Buckingham theorem. Furthermore, in the absence of knowledge of the true solution, in the presence of noise, and the face of data scarcity, such an improvement is far from guaranteed.

3.5 Neural Networks with and without Dimensional Analysis

The previous experiments demonstrate the impact of dimensional analysis in sparse regression. Here, we investigate the impact of nondimensionalization of input parameters on the performance of deep learning models. Specifically, we study the ability of a NN to recover Stokes' law for flow past a sphere (Section 3.1). The fully connected feedforward NN comprises four hidden layers with 20 neurons in each layer. It uses a tanh activation function in all layers except the last one, is initialized via Xavier initialization (Glorot and Bengio, 2010), and employs the LBFGS optimizer with the learning rate of 0.1 for training. Two NN-based strategies are considered. The first involves four inputs representing the dimensional quantities constituting the Reynolds number Re , with drag coefficient C_d as the output. The second has a single input (Reynolds number Re) and one output (drag coefficient C_d). For each input-output pair in the training domain, we

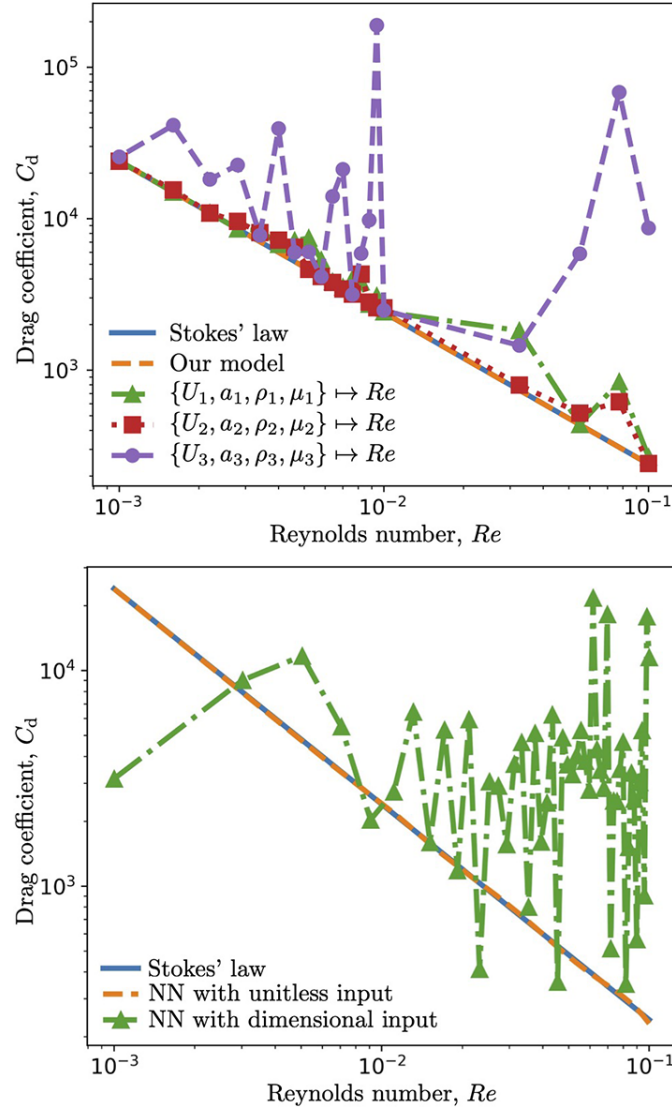


FIG. 3: Relationship between drag coefficient for a sphere, C_d , and Reynolds number Re predicted by (top) the lasso regression and (bottom) neural networks, with and without dimensional analysis of data for three different combinations of the model parameters $\{U, a, \rho, \mu\}$ resulting in the same Reynolds number Re . The disparity between the latter three predictions violates the system's symmetry, with all predictions deviating from Stokes' law, $C_d = 24/Re$, (solid line).

randomly generated 200 training points, as specified above. To have a fair comparison, 2500 epochs are carried out to train both NNs. The test data are generated in the manner identical to that used in the previous section.

Figure 3 illustrates the failure of the NN trained on the raw data $\{\mathcal{S}_i; C_{d,i}\}$, and the success of the NN trained on unitless data, to match Stokes' law. While one could optimize the NN performance to minimize the overfitting, the former NN struggles to generalize to unseen

inputs during training. This experiment underscores that irrespective of the chosen method—whether regression-based or NN-based—the model that ignores the physical symmetry fails to generalize. Conversely, the model trained on unitless data generalizes effectively on the test data or an out-of-distribution set.

4. CONCLUSIONS

Our numerical experiments demonstrate that dimensional analysis of data is a necessary first step in ML-assisted discovery of constitutive relations and secondary laws. When combined with sparse (e.g., lasso) regression, dimensional analysis yields parsimonious models, e.g., for drag coefficients, which are explainable and generalizable to parametric regimes not seen in training. As an example, we presented a new model for the drag coefficient for creeping flow past an inclined prolate ellipsoid, which, in various limits, reduces approximately to the classic relations, such as Stokes' law for a sphere or the Happel–Brenner solution for an ellipsoid aligned with the flow direction.

The failure to subject the training data to dimensional analysis is likely to yield models that might fit the test data well and, yet, give nonphysical predictions. Specifically, such ML models, e.g., sparse regression or deep neural networks, are likely to ignore the fundamental symmetries present in any physical system. We demonstrated this failure on a relatively simple problem of creeping flow past a sphere.

Our study focused on the ability of the Buckingham Π theorem to enhance the generalizability of ML-discovered secondary laws (constitutive relations) and to enforce their symmetry-preserving features. Several related issues are left for future research. The dimensional analysis alone might have to be supplemented with expert knowledge in order to choose correct variables or quantities of interest (Bakarji et al., 2022). The Buckingham theorem does not deal with unitless inputs, e.g., porosity, yet their amalgamation into dimensionless groups could improve the generalizability of a learned model. To sum up, the choice of the unitless groups is nonunique and subjective, requiring the application of human intelligence and a prior understanding of the physical system.

The codes and data to reproduce the results of this paper can be found at https://github.com/chandratue/Phy_in_PIML.

ACKNOWLEDGMENTS

A.C. thanks the Information and Knowledge Society fellowship from Université de Lille, France for supporting his studies. J.B. thanks the Center for Advanced Mathematical Sciences (CAMS) at the American University of Beirut, for supporting this collaboration. D.M.T. was supported in part by the Air Force Office of Scientific Research under grant FA9550-21-1-0381, by the National Science Foundation under award 2100927, and by the Office of Advanced Scientific Computing Research (ASCR) within the Department of Energy Office of Science under award number DE-SC0023163.

REFERENCES

Adcock, B., Multivariate Modified Fourier Series and Application to Boundary Value Problems, *Numer. Math.*, vol. **115**, pp. 511–552, 2010.

- Andersson, H. and Jiang, F., Forces and Torques on a Prolate Spheroid: Low-Reynolds-Number and Attack Angle Effects, *Acta Mech.*, vol. **230**, pp. 431–447, 2019.
- Bakarji, J., Callaham, J., Brunton, S., and Kutz, J., Dimensionally Consistent Learning with Buckingham Pi, *Nature Comput. Sci.*, vol. **2**, pp. 1–11, 2022.
- Bakarji, J. and Tartakovsky, D.M., Data-Driven Discovery of Coarse-Grained Equations, *J. Comput. Phys.*, vol. **434**, p. 110219, 2021.
- Brunton, S., Noack, B., and Koumoutsakos, P., Machine Learning for Fluid Mechanics, *Annu. Rev. Fluid Mech.*, vol. **52**, pp. 477–508, 2020.
- Chandra, A., Daniels, B., Curti, M., Tiels, K., Lomonova, E.A., and Tartakovsky, D.M., Discovery of Sparse Hysteresis Models for Piezoelectric Materials, *Appl. Phys. Lett.*, vol. **122**, no. 21, 2023a.
- Chandra, A., Kapoor, T., Daniels, B., Curti, M., Tiels, K., Tartakovsky, D.M., and Lomonova, E.A., Neural Oscillators for Magnetic Hysteresis Modeling, arXiv preprint arXiv:2308.12002, 2023b.
- Fukami, K. and Taira, K., Robust Machine Learning of Turbulence through Generalized Buckingham Pi-Inspired Pre-Processing of Training Data, *APS Division of Fluid Dynamics Meeting Abstracts*, Phoenix, AZ, pp. A31–004, 2021.
- Glorot, X. and Bengio, Y., Understanding the Difficulty of Training Deep Feedforward Neural Networks, in *Proc. of the 13th Int. Conf. on Artificial Intelligence and Statistics*, Sardinia, Italy, pp. 249–256, 2010.
- Happel, J. and Brenner, H., *Low Reynolds Number Hydrodynamics: With Special Applications to Particulate Media*, Berlin: Springer Science & Business Media, 1983.
- Hölzer, A. and Sommerfeld, M., New Simple Correlation Formula for the Drag Coefficient of Non-Spherical Particles, *Powder Technol.*, vol. **184**, no. 3, pp. 361–365, 2008.
- Jofre, L., del Rosario, Z., and Iaccarino, G., Data-Driven Dimensional Analysis of Heat Transfer in Irradiated Particle-Laden Turbulent Flow, *Int. J. Multiphase Flow*, vol. **125**, p. 103198, 2020.
- Kapoor, T., Wang, H., Núñez, A., and Dollevoet, R., Predicting Traction Return Current in Electric Railway Systems through Physics-Informed Neural Networks, in *2022 IEEE Symposium Series on Computational Intelligence (SSCI)*, Singapore, pp. 1460–1468, 2022.
- Kapoor, T., Chandra, A., Tartakovsky, D., Wang, H., Nunez, A., and Dollevoet, R., Neural Oscillators for Generalizing Parametric PDEs, in *The Symbiosis of Deep Learning and Differential Equations III*, New Orleans, LA, 2023a.
- Kapoor, T., Wang, H., Núñez, A., and Dollevoet, R., Physics-Informed Machine Learning for Moving Load Problems, arXiv preprint arXiv:2304.00369, 2023b.
- Kapoor, T., Wang, H., Núñez, A., and Dollevoet, R., Physics-Informed Neural Networks for Solving Forward and Inverse Problems in Complex Beam Systems, *IEEE Trans. Neural Networks Learn. Syst.*, vol. **35**, no. 5, pp. 5981–5995, 2023c.
- Kapoor, T., Chandra, A., Tartakovsky, D.M., Wang, H., Nunez, A., and Dollevoet, R., Neural Oscillators for Generalization of Physics-Informed Machine Learning, in *Proc. of the AAAI Conf. on Artificial Intelligence*, Vancouver, BC, vol. **38**, pp. 13059–13067, 2024a.
- Kapoor, T., Wang, H., Núñez, A., and Dollevoet, R., Transfer Learning for Improved Generalizability in Causal Physics-Informed Neural Networks for Beam Simulations, *Eng. Appl. Artif. Intel.*, vol. **133**, p. 108085, 2024b.
- Karniadakis, G., Kevrekidis, I., Lu, L., Perdikaris, P., Wang, S., and Liu, Y., Physics-Informed Machine Learning, *Nat. Rev. Phys.*, vol. **3**, p. 422–440, 2021.
- Kim, J., Lee, K., Lee, D., Jhin, S.Y., and Park, N., DPM: A Novel Training Method for Physics-Informed Neural Networks in Extrapolation, in *Proc. of the AAAI Conf. on Artificial Intelligence*, Virtual, vol. **35**, pp. 8146–8154, 2021.
- Kochkov, D., Smith, J., Alieva, A., Wang, Q., Brenner, M., and Hoyer, S., Machine Learning–Accelerated Computational Fluid Dynamics, *Proc. Natl. Acad. Sci. U.S.A.*, vol. **118**, no. 21, p. e2101784118, 2021.

- Lennon, K., McKinley, G., and Swan, J., Scientific Machine Learning for Modeling and Simulating Complex Fluids, *Proc. Natl. Acad. Sci.*, vol. **120**, no. 27, p. e2304669120, 2023.
- Livi, C., Di Staso, G., Clercx, H., and Toschi, F., Drag and Lift Coefficients of Ellipsoidal Particles under Rarefied Flow Conditions, *Phys. Rev. E*, vol. **105**, no. 1, p. 015306, 2022.
- Lu, H. and Tartakovsky, D.M., Prediction Accuracy of Dynamic Mode Decomposition, *SIAM J. Sci. Comput.*, vol. **42**, no. 3, pp. A1639–A1662, 2020.
- Matchev, K.T., Matcheva, K., and Roman, A., Analytical Modeling of Exoplanet Transit Spectroscopy with Dimensional Analysis and Symbolic Regression, *Astrophys. J.*, vol. **930**, no. 1, 2022. DOI: 10.3847/1538-4357/ac610c
- Oppenheimer, M., Doman, D., and Merrick, J., Multi-Scale Physics-Informed Machine Learning Using the Buckingham Pi Theorem, *J. Comput. Phys.*, vol. **474**, p. 111810, 2023.
- Ouchene, R., Khalij, M., Arcen, B., and Tanière, A., A New Set of Correlations of Drag, Lift and Torque Coefficients for Non-Spherical Particles and Large Reynolds Numbers, *Powder Technol.*, vol. **303**, pp. 33–43, 2016.
- Schmelzer, M., Dwight, R., and Cinnella, P., Discovery of Algebraic Reynolds-Stress Models Using Sparse Symbolic Regression, *Flow Turbul. Combust.*, vol. **104**, pp. 579–603, 2020.
- Sharma, P., Chung, W., Akoush, B., and Ihme, M., A Review of Physics-Informed Machine Learning in Fluid Mechanics, *Energies*, vol. **16**, p. 2343, 2023.
- Souta, M., Molina, J., and Takashi, T., Machine-Learned Constitutive Relations for Multi-Scale Simulations of Well-Entangled Polymer Melts, *Phys. Fluids*, vol. **35**, p. 063113, 2023.
- Taghizadeh, S., Witherden, F.D., and Girimaji, S.S., Turbulence Closure Modeling with Data-Driven Techniques: Physical Compatibility and Consistency Considerations, *New J. Phys.*, vol. **22**, no. 9, p. 093023, 2020.
- Tibshirani, R., Regression Shrinkage and Selection via the Lasso, *J. R. Stat. Soc.: Ser. B*, vol. **58**, no. 1, pp. 267–288, 1996.
- Villar, S., Yao, W., Hogg, D., Blum-Smith, B., and Dumitrascu, B., Dimensionless Machine Learning: Imposing Exact Units Equivariance, *J. Mach. Learn. Res.*, vol. **24**, pp. 1–32, 2023.
- Wang, R., Walters, R., and Yu, R., Incorporating Symmetry into Deep Dynamics Models for Improved Generalization, in *Int. Conf. on Learning Representations*, 2021.
- Xie, X., Samaei, A., Guo, J., Liu, W., and Gan, Z., Data-Driven Discovery of Dimensionless Numbers and Governing Laws from Scarce Measurements, *Nat. Commun.*, vol. **13**, no. 1, p. 7562, 2022.
- Yarin, L., *The Pi-Theorem: Applications to Fluid Mechanics and Heat and Mass Transfer*, Berlin: Springer-Verlag, 2012.