Artificial neural network based response surface for data-driven dimensional analysis

Zhaoyue Xu [a,b], Xinlei Zhang [a,b], Shizhao Wang [a,b,*], Guowei He [a,b]

[a] *The State Key Laboratory of Nonlinear Mechanics, Institute of Mechanics, Chinese Academy of Sciences, Beijing 100190, China*
[b] *School of Engineering Sciences, University of Chinese Academy of Sciences, Beijing 101408, China*

A R T I C L E   I N F O

A B S T R A C T

The classical dimensional analysis method has limitations in determining the uniqueness and relative importance of the dimensionless quantities. A machine-learning based dimensional analysis method is proposed to address the limitations. The proposed method identifies *unique* and *relevant* dimensionless quantities by combining an artificial neural network with the data-driven dimensional analysis. We employ a fully connected neural network to construct the ridge function for the response surface in a physical system. The gradient of the response surface for active subspace analysis is computed based on a finite difference approximation. An effective approach is proposed to determine the independent variables of experimental measurements or numerical simulations for computing the gradient of the response surface. The proposed method is validated by analyzing benchmark pipe flows and a fluid-structure interaction system. The dominant dimensionless quantities obtained by the proposed method are consistent with those reported in the literature. The proposed method has the advantage of identifying the relatively important dimensionless quantities without referring to the complex theoretical equations.

1. Introduction

Dimensional analysis investigates the relationships between quantities based on their dimensions, which is an effective way to study complex physical systems. The premise of dimensional analysis is that these relationships do not change when the units of measurement are altered [1,2]. The dimensional analysis has been utilized to identify the dimensionless quantities and reduce complex physical systems to simpler forms [3,4]. The specific dimensionless quantities can describe the scale-free properties of different physical systems. The physical phenomena described by equal dimensionless quantities are similar, which suggests that the results of model experiment or simulation can be utilized to reproduce interested phenomena [5].

The well-known Buckingham's Pi theorem summarizes the central theory of the classical dimensional analysis method [3,6]. According to this theorem, the number of dimensionless quantities needed to define a system equals the total number of dimensional quantities in the system, m, minus the number of fundamental quantities with independent dimensions for the system, k, i.e., the system can be described with $(m - k)$ dimensionless quantities. This is a type of dimension reduction approach by which the number of variables to describe the system decreases. The classical dimensional analysis method

---

* Corresponding author.
  E-mail address: wangsz@lnm.imech.ac.cn (S. Wang).

experiences two major limitations: (1) the dimensionless quantities are not unique, and (2) the relative importance of the different dimensionless quantities is not measured [1,4].

To address the two major limitations of the classical dimensional analysis method, Constantine et al. [1] proposed a data-driven dimensional analysis methodology to identify *unique* and *relevant* dimensionless quantities.

The basic idea of their methodology relies on the connection between the dimension and vector space. Specifically, their data-driven dimensional analysis methodology combines the classical dimensional analysis method with the concepts of the ridge function and the active subspace [7]. Constantine et al. [1] found that the expression for the dimensionless quantities in a physical system can be modeled with a ridge function. Given any ridge function, we can find certain specific directions along which the dependent variables change dramatically. The low-dimensional subspace of these specific directions is the active subspace [8].

The active subspace in dimensional analysis represents the relatively important dimensionless quantities in the physical system. Furthermore, the active subspace is unique, because the basis of active space is unique and the directions of basis are corresponding to the exponent of dimensionless quantities. The unique active subspace indicates the uniqueness of the dimensionless quantities.

In data-driven dimensional analysis [1], a dimension matrix is first obtained according to the classical dimensional analysis, and a response surface is then constructed to determine the active subspace, which helps to identify the new dimensionless *unique* and *relevant* quantities. One of the crucial steps of the data-driven dimensional analysis method is the construction of the response surface and the computation of its gradient. There are extensive studies on response surface methodologies [9,10].

Two algorithms have been proposed in the work of Constantine et al. [1] to construct and compute the gradient of response surface. One algorithm is based on the finite difference-based method, where the independent quantities are obtained from experiments or numerical simulations at specific intervals to ensure numerical calculation of the response surface gradient in the active subspace. The other algorithm is a data-driven algorithm which applies Gaussian process regression to fit the response surface and compute its gradient. Both of these two algorithms have been successfully used in analyzing the unique dimensionless quantities in laminar and turbulent pipe flows. Jofre et al. [4] further improved the data-driven algorithm and applied it to an irradiated particle-laden turbulence case. Their results show that the data-driven algorithm can accurately identify the dimensionless quantities that determine heat transfer in irradiated particle-laden turbulent flow. Yang et al. [11] built a global polynomial chaos response model and computed gradients with this model, by which they efficiently solved some high-dimensional problems. Fukumizu et al. [12] established a kernel-based approximation to compute the gradient. This method theoretically guarantees dimension reduction in regression with a low computational time cost. Constantine et al. [13] further applied sliced average variance estimation (SAVE) to calculate the direction of the response surface to reduce the physical dimensions.

With the development of computer technology and algorithms, neural networks have been successfully employed by researchers in data assimilation [14], turbulence modeling [15–17], subgrid model for large eddy simulation [18,19], drag reduction [20], velocity predicting [21] and unstructured mesh modeling [22]. In recent years, construction of the response surface via neural networks has been demonstrated an effective and feasible approach [23]. Neural network has shown its advantages in the construction of the response surface because [24] (1) it can process highly nonlinear complex systems with the ability to fit and predict data, (2) new data can be added while the neural network remains unchanged. Considering that the neural networks provide an alternative approach to effectively predict the response surface, it is attractive to combine the neural networks with data-driven dimensional analysis.

The aim of this work is to introduce a machine-learning based data-driven dimensional analysis method to identify the *unique* and *relevant* dimensionless quantities. We use a feedforward neural network to construct the response surface and compute its gradient, and then identify the dominant dimensionless quantities by estimating the active subspace. The gradient of the response surface for active subspace analysis is computed based on a finite difference approximation. An effective approach is proposed to determine the independent variables of experimental measurements or numerical simulations for computing the finite difference. The proposed method is validated by analyzing the pressure drop in benchmark pipe flows and the drag experienced by a flexible body in flows.

The remainder part of this paper is organized as follows. The theory and algorithm of the dimensional analysis method and the neural network based response surface are described in Section 2. The proposed method is validated by analyzing benchmark pipe flows in Section 3. Then, the proposed method is employed to analyze the drag acting on a flexible body in Section 4. Finally, conclusions are drawn in Section 5.

2. Machine-learning based data-driven dimensional analysis

2.1. Data-driven dimensional analysis

According to the classical dimensional analysis, the dimensions of m quantities $\mathbf{q} = [q_1, \cdots, q_m]^T$ in a system can be expressed as products of k fundamental dimensions. The dimension vector of the i-th quantity q_i, denoted as $\mathbf{v}_d(q_i)$, is a function that returns the exponents of the dimensions of this quantity. For instance, if q_i is the velocity u in a physical system with three fundamental dimensions ($k = 3$), $[L]$, $[T]$ and $[M]$, the dimension of q_i is $[L]^1[T]^{-1}[M]^0$, and the dimension vector is $\mathbf{v}_d(q_i) = [1, -1, 0]^T$. In the case of m independent quantities in a physical system with k fundamental dimensions,

a matrix $\mathbf{D} = [\mathbf{v}_d(q_1), \cdots, \mathbf{v}_d(q_m)]$ of $k \times m$ elements contains the dimension vectors of m independent quantities. We denote a matrix $\mathbf{W} = [\mathbf{w}_1, \cdots, \mathbf{w}_n] \in \mathbb{R}^{m \times n}$ as the null space of the matrix $\mathbf{D}$

$$\mathbf{DW} = [0]_{k \times n}, \tag{1}$$

where $n = m - k$. In general, we can compute the matrix $\mathbf{W}$ via linear algebra methods or via classical dimensional analysis with Schmidt orthogonalization. After we determine the matrix $\mathbf{W}$, the dimensionless quantities $\mathbf{\Pi} = [\pi_1, \cdots, \pi_n]^T$ containing the n dimensionless independent quantities can be obtained. For the i-th dimensionless independent quantity π_i in $\mathbf{\Pi} = [\pi_1, \cdots, \pi_n]^T$, we have the expression as follows,

$$
\begin{aligned}
\pi_i &= q_1{}^{w_{1i}} \cdots q_m{}^{w_{mi}} \\
&= \exp\left(\log\left(q_1{}^{w_{1i}} \cdots q_m{}^{w_{mi}}\right)\right) \\
&= \exp\left(\sum_{j=1}^{m} w_{ji} \log\left(q_j\right)\right) \\
&= \exp\left(\mathbf{w}_i{}^T \log(\mathbf{q})\right),
\end{aligned}
\tag{2}
$$

where $\mathbf{q} = [q_1, \cdots, q_m]^T$ denotes the independent quantities in the physical system, w_{ji} is an element of the matrix $\mathbf{W}$ and $\mathbf{w}_i$ is the i-th column of the matrix $\mathbf{W}$. It is noted that there are some controversies over the dimensional quantities involving transcendental functions [25,26]. In accordance with the work of Constantine et al. [1], we use the expression as Eq. (2) to formally represent the products of powers of physical quantities.

The dimensionless dependent quantity π_t of the system can be expressed as a function of the dimensionless independent quantities $\mathbf{\Pi} = [\pi_1, \cdots, \pi_n]^T$,

$$\pi_t = f(\pi_1, \cdots, \pi_n). \tag{3}$$

Based on Eq. (2) and Eq. (3), we obtain

$$
\begin{aligned}
\pi_t &= f(\pi_1, \cdots, \pi_n) \\
&= f(\exp(\mathbf{w}_1{}^T \log(\mathbf{q})), \cdots, \exp(\mathbf{w}_n{}^T \log(\mathbf{q}))).
\end{aligned}
\tag{4}
$$

We define $\mathbf{x} = \log(\mathbf{q})$ and rewrite Eq. (4) as follows,

$$
\begin{aligned}
\pi_t &= f(\exp(\mathbf{w}_1{}^T \mathbf{x}), \cdots, \exp(\mathbf{w}_n{}^T \mathbf{x})) \\
&= g(\mathbf{w}_1{}^T \mathbf{x}, \cdots, \mathbf{w}_n{}^T \mathbf{x}) \\
&= g\left(\mathbf{W}^T \mathbf{x}\right),
\end{aligned}
\tag{5}
$$

where g is a ridge function [27,28].

Given a ridge function $f_r : \mathbb{R}^m \to \mathbb{R}$ with independent variables $\mathbf{x}$, we define

$$f_r(\mathbf{x}) = g\left(\mathbf{W^T x}\right). \tag{6}$$

For any vector $\mathbf{J}$ orthogonal to the columns of $\mathbf{W}$, $\mathbf{W}^T \mathbf{J} = 0$, we have

$$f_r(\mathbf{x}) = g\left(\mathbf{W^T}(\mathbf{x})\right) = g\left(\mathbf{W^T}(\mathbf{x} + \mathbf{J})\right) = f_r(\mathbf{x} + \mathbf{J}). \tag{7}$$

In other words, the ridge function $f_r(\mathbf{x})$ remains constant when the variables change along the direction orthogonal to the columns of $\mathbf{W}$. The variables that change along particular directions do not affect the function value, and we can reduce the dimensions of the variables along these directions. For example, the function $y = e^{2x_1 + 3x_2}$ is a typical ridge function. The dependent variable y varies rapidly when the independent variable $\mathbf{x}$ changes along the direction $(2, 3)$, and y remains constant when the independent variable changes along the orthogonal direction $(-3, 2)$. This feature generalizes the idea of identifying a subset of dominant variables, since the ridge function exhibits dominant directions. We aim to focus on those directions that are dominant and ignore the directions where the ridge function remains constant. This is the motivation behind the active subspace [7].

We can define a $n \times n$ symmetric and positive semidefinite matrix $\mathbf{C}$ based on the continuous differential function $g : \mathbb{R}^n \to \mathbb{R}$ as

$$\mathbf{C} = \int \nabla g(\mathbf{x}_r) \nabla g(\mathbf{x}_r)^T \sigma(\mathbf{x}_r) d\mathbf{x}_r, \tag{8}$$

where $\mathbf{x}_r \in \mathbb{R}^n$ is a vector distributed according to the joint density σ, $\sigma : \mathbb{R}^n \to \mathbb{R}_+$ is a weight function and $\nabla g \in \mathbb{R}^n$ is the gradient of ridge function g. Matrix $\mathbf{C}$ is the uncentered covariance of the gradient vector [8], which enables eigenvalue decomposition

$$\mathbf{C} = \mathbf{S}\boldsymbol{\Lambda}\mathbf{S^T}, \tag{9}$$

where $\mathbf{S} = [\mathbf{s}_1, \cdots, \mathbf{s}_n]$ is the orthogonal matrix of the eigenvectors, $\mathbf{s}_n$ is the n-th column eigenvector, and $\boldsymbol{\Lambda}$ is the diagonal matrix of n nonnegative eigenvalues. For the i-th eigenvalue λ_i in $\boldsymbol{\Lambda}$, the Rayleigh quotient form of the eigenvalue reveals the expression as follows,

$$\begin{aligned}
\lambda_i = \mathbf{s}_i^{\mathbf{T}}\mathbf{C}\mathbf{s}_i &= \mathbf{s}_i^{\mathbf{T}} \left(\int \nabla g(\mathbf{x}_r) \nabla g(\mathbf{x}_r)^T \sigma(\mathbf{x}_r) d\mathbf{x}_r \right) \mathbf{s}_i \\
&= \int \left(\nabla g(\mathbf{x}_r)^T \mathbf{s}_i \right)^2 \sigma(\mathbf{x}_r) d\mathbf{x}_r,
\end{aligned} \tag{10}$$

where $\mathbf{s}_i$ denotes the corresponding eigenvector. Eq. (10) indicates that λ_i must be greater than or equal to 0, and this parameter measures the average rate of change along the eigenvector direction. If $\lambda_i > \lambda_j$, the change in ridge function g along the direction of $\mathbf{s}_i$ occurs faster than that along the direction of $\mathbf{s}_j$. Therefore, we sort the eigenvalues in descending order, $\lambda_1 > \cdots > \lambda_p > \lambda_{p+1} \cdots > \lambda_n$. If the first p eigenvalues are much greater than the $(p + 1)$-st eigenvalue ($\lambda_p \gg \lambda_{p+1}$), the active subspace is then the eigenvector space $[\mathbf{s}_1, \cdots, \mathbf{s}_p]$.

We introduce a new set of variables for the dimensionless relationship Eq. (3)

$$\begin{aligned}
\pi_t &= f(\pi_1, \cdots, \pi_n) \\
&= f(\exp(\log(\pi_1)), \cdots, \exp(\log(\pi_n))) \\
&= g(\log(\pi_1), \cdots, \log(\pi_n)) \\
&= g(\gamma_1, \cdots, \gamma_n),
\end{aligned} \tag{11}$$

where $\gamma_i = \log(\pi_i)$. We thus obtain a vector $\boldsymbol{\Gamma} = [\gamma_1, \cdots, \gamma_n]^T$, according to Eq. (5) and $\mathbf{x} = \log(\mathbf{q})$

$$\boldsymbol{\Gamma} = \mathbf{W^T}\mathbf{x} = \mathbf{W^T}\log(\mathbf{q}). \tag{12}$$

Take $g(\boldsymbol{\Gamma})$ into Eq. (8), we have,

$$\int \nabla g(\boldsymbol{\Gamma}) \nabla g(\boldsymbol{\Gamma})^T \sigma(\boldsymbol{\Gamma}) d\boldsymbol{\Gamma} = \mathbf{S}\boldsymbol{\Lambda}\mathbf{S^T} = \mathbf{C}. \tag{13}$$

The vector $\boldsymbol{\Gamma}$ can be adjusted toward the direction of the eigenvectors as follows,

$$\mathbf{S^T}\boldsymbol{\Gamma} = \mathbf{S^T}\mathbf{W^T}\mathbf{x} = \mathbf{Z^T}\mathbf{x}, \tag{14}$$

where $\mathbf{Z} = \mathbf{W}\mathbf{S} \in \mathbb{R}^{m \times n}$. Recall that n is the number of dimensionless quantities and m is the number of independent variables. The columns of $\mathbf{Z}$ define new dimensionless quantities that form the unique and associated dimensionless quantities $\hat{\boldsymbol{\Pi}} = [\hat{\pi}_1, \cdots, \hat{\pi}_n]^T$. In other words, each component in $\hat{\boldsymbol{\Pi}} = [\hat{\pi}_1, \cdots, \hat{\pi}_n]^T$ can be expressed as

$$\hat{\pi}_i = \exp\left(\mathbf{z}_i^T \log(\mathbf{q}) \right), \tag{15}$$

where $\mathbf{z}_i^T$ is the i-th row of matrix $\mathbf{Z^T}$. Then, the active subspace of the dimensionless quantities can be constructed based on the eigenvalue diagonal matrix $\boldsymbol{\Lambda}$. Through the active subspace theory, we can determine the relative importance of the dimensionless quantities in $\hat{\boldsymbol{\Pi}}$.

2.2. Machine-learning based data-driven dimensional analysis method

We propose a machine-learning based data-driven dimensional analysis in this subsection to demonstrate the feasibility of neural networks in constructing and computing the response surface. We use a fully connected feedforward neural network (hereafter referred to as the neural network in this work), to construct a response surface and compute its gradient in order to conduct machine-learning based data-driven dimensional analysis. The schematics of the neural network are shown in Fig. 1, which consists of an input layer, an output layer and multiple hidden layers. Neurons are distributed in each layer. With adequate hidden layers, they can approximate continuous complex functions [29].

The collection of neurons in a specific layer in the neural network can be considered a vector. The input vector is denoted as $\mathbf{x}$, and the output vector is denoted as $\mathbf{y}$. We denote the output of the i-th layer as $\mathbf{H}_i$. The neural network in the form of b hidden layers can be expressed as follows,
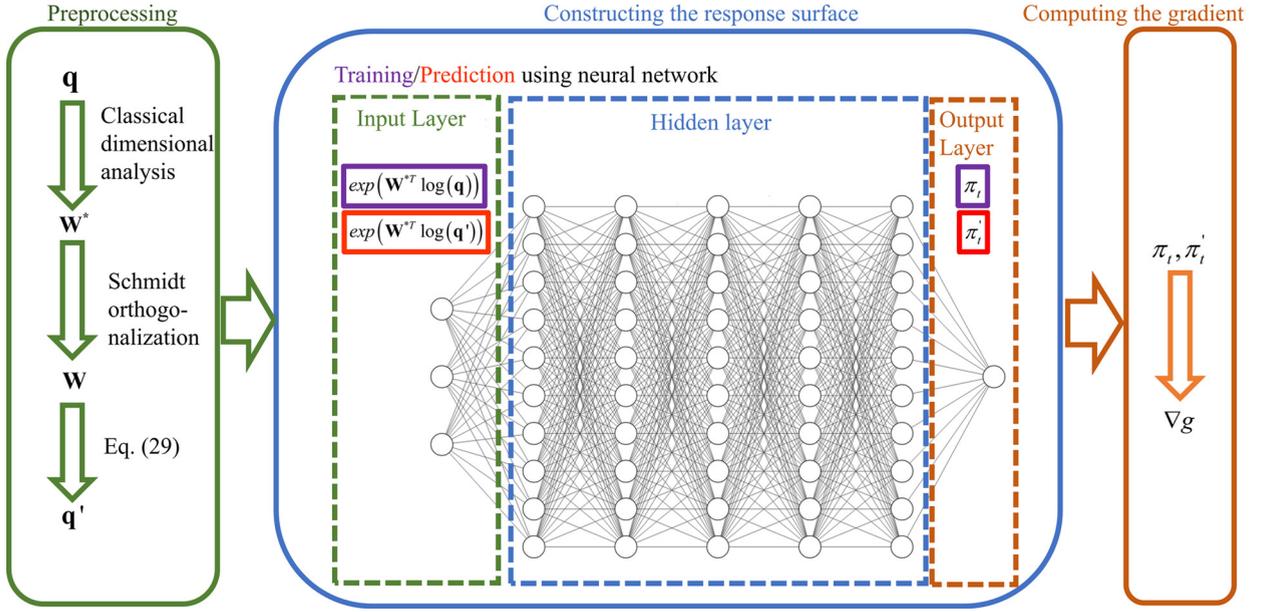
Fig. 1. Schematics of computing the gradient of response surface g by using fully connected neural network. The computation of the gradient includes three stages: preprocessing, constructing the response surface, and computing the gradient. In the preprocessing stage, we start from the physical quantities $\mathbf{q}$ to compute $\mathbf{W}^*$ by classical dimensional analysis. The matrix $\mathbf{W}$ can be obtained via Schmidt orthogonalization of $\mathbf{W}^*$. The input quantities $\mathbf{q}'$ are computed by Eq. (29). In the stage of construing the response surface, the response surface is constructed using neural network. For example, the neural network shown in this figure is the network utilized in analyzing the drag of a flexible body in Section 4. The input layer for this case consists of three circles on the left-hand-side of the network. The circle on the right-hand-side represents the output layer, which outputs the dependent variable π_t. The hidden layers are located between the input layer and output layer. In the third stage, the gradient of response surface is computed by Eq. (20), where π_t' are predictions of the neural network. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

$$
\begin{cases}
\mathbf{H}_0 &= \mathbf{x}, \\
\mathbf{H}_{i+1} &= \phi\left(\mathbf{C}_i + \mathbf{R}_i \mathbf{H}_i\right), 0 \le i \le b-1 \\
\mathbf{y} &= f_\theta(\mathbf{x}; \boldsymbol{\theta}) = \mathbf{R}_b \mathbf{H}_b + \mathbf{C}_b, \boldsymbol{\theta} := \{\mathbf{R}_i, \mathbf{C}_i\}_{i=0}^b,
\end{cases}
\tag{16}
$$

where $\mathbf{H}_i$ is the output of i-th layer, $\mathbf{C}_i$ is the output offset vector, $\mathbf{R}_i$ is the i-th column of the weight matrix $\mathbf{R}$, and ϕ is the nonlinear neural network activation function. In this work, rectified linear unit (ReLU) and hyperbolic tangent $tanh(x)$ are chosen as activation functions $\phi(x)$. Vector $\boldsymbol{\theta}$ collects all the undetermined neural network parameters $\mathbf{C}_i$ and $\mathbf{R}_i$. The training process of the neural network utilizes the loss function and optimizer to adjust $\boldsymbol{\theta}$.

The loss function describes the scalar *loss* associated with each labeled training data points $(\mathbf{x}_{train}, \mathbf{y}_{train})$. We use the regression function as follows,

$$
\sum_{i=1}^{N} \left\| \mathbf{y}_{train}^i - f_\theta(\mathbf{x}_{train}^i; \boldsymbol{\theta}) \right\|^2,
\tag{17}
$$

where N is the number of training sets. The optimizer employed in this work is the Adam method [30], which considers first and second moment estimations of the gradient. In this work, the inputs are dimensionless quantities $\boldsymbol{\Pi}$, and the output is the dimensionless quantity π_t. We normalize the input parameters to ensure that any parameters with relatively large values do not disproportionately affect the results [7]. The parameters are normalized as follows. For a set of parameters $(\alpha_1, \cdots, \alpha_i, \cdots, \alpha_n)$ with the lower bounds α_l and the upper bounds α_u. The normalized parameters $(\hat{\alpha}_1, \cdots, \hat{\alpha}_i, \cdots, \hat{\alpha}_n)$ are computed by $\hat{\alpha}_i = 2(\alpha_i - \alpha_l)/(\alpha_u - \alpha_l) - 1, i = 1 \cdots n$.

The data are randomly divided into training and validation sets. The *loss* of the validation set (*val_loss*) is the overfitting criterion.

The implementation of the machine-learning based data-driven dimensional analysis method proposed in this work is given as follows:

1. Compute the matrix $\mathbf{W}$ from a set of data $\left(\left(\pi_t^{(1)}, \mathbf{q}^{(1)}\right), \cdots, \left(\pi_t^{(N)}, \mathbf{q}^{(N)}\right)\right)$ obtained from N experimental measurements or numerical simulations, where $\mathbf{q}$ denote the dimensional quantities (independent variables). Here, π_t is the dimensionless quantity of interest (dependent variable).

   The dimensionless independent quantities can be derived from the dimensional quantities $\mathbf{q}$ by using the classical dimensional analysis and are employed to compute an auxiliary matrix $\mathbf{W}^*$, whose columns consist of the exponents of

the dimensional quantities for the dimensionless quantities. Then, the matrix $\mathbf{W}$ in Eq. (1) can be obtained via Schmidt orthogonalization of matrix $\mathbf{W}^*$.

2. Compute the dimensionless quantities $\mathbf{\Gamma}^{(j)} = \left[\gamma_1{}^{(j)}, \cdots, \gamma_n{}^{(j)} \right] (j = 1, 2, \ldots, N)$ as follows,

$$\gamma_i{}^{(j)} = \mathbf{w}_i{}^T \log \left(\mathbf{q}^{(j)} \right), i = 1 \ldots n. \tag{18}$$

3. Construct the response surface $g(\mathbf{\Gamma})$ from $\mathbf{\Gamma}^{(j)}$ by using the neural network,

$$\pi_t^{(j)} \approx g \left(\mathbf{\Gamma}^{(j)} \right) \approx f_\theta \left(exp \left(\mathbf{W}^{*T} \log \left(\mathbf{q}^{(j)} \right) \right) \right). \tag{19}$$

4. Compute the gradient of the response surface $g(\mathbf{\Gamma})$ based on the trained neural network as follows (see more details in Section 2.3),

$$\frac{\partial g(\mathbf{\Gamma}^{(j)})}{\partial \gamma_i} = \frac{\pi_{ti}^{\prime(j)} - \pi_t^{(j)}}{\Delta h} \tag{20}$$

$$\pi_{ti}^{\prime(j)} \approx g \left(\left[\gamma_1{}^{(j)}, \cdots, \gamma_i{}^{(j)} + \Delta h, \cdots, \gamma_n{}^{(j)} \right] \right). \tag{21}$$

5. Evaluate the active subspace of $g(\mathbf{\Gamma})$ by computing the uncentered covariance of the gradient vector of $g(\mathbf{\Gamma})$,

$$\mathbf{C} = \int \nabla g(\mathbf{\Gamma}) \nabla g(\mathbf{\Gamma})^T \sigma(\mathbf{\Gamma}) d\mathbf{\Gamma} = \mathbf{S} \mathbf{\Lambda} \mathbf{S}^T. \tag{22}$$

6. Compute the new dimensionless quantities $\hat{\mathbf{\Pi}} = [\hat{\pi}_1, \cdots, \hat{\pi}_n]^T$ as follows,

$$\mathbf{Z} = \mathbf{W} \mathbf{S} \tag{23}$$

$$\hat{\pi}_i = \exp \left(\mathbf{z}_i^T \log(\mathbf{q}) \right). \tag{24}$$

Based on the above steps, we can obtain the *unique* and *relevant* dimensionless quantities $\hat{\mathbf{\Pi}} = [\hat{\pi}_1, \cdots, \hat{\pi}_1]^T$ of the physical system.

As pointed out by Constantine et al. [1], different numerical quadrature methods can be used in numerically computing Eq. (22). We use Monte Carlo method to compute the numerical quadrature in Eq. (22) in accordance with the work of Constantine et al. [1]. For the set of data $\left(\pi_t^{(j)}, \mathbf{q}^{(j)} \right)$, $(j = 1, \cdots, N)$ given in Step 1, Eq. (22) can be approximated by

$$\mathbf{C} \approx \frac{1}{N} \sum_{j=1}^{N} \nabla g(\mathbf{\Gamma}^{(j)}) \nabla g(\mathbf{\Gamma}^{(j)})^T \tag{25}$$

where $\nabla g(\mathbf{\Gamma}^{(j)}) = \left[\frac{\partial g(\mathbf{\Gamma}^{(j)})}{\partial \gamma_1}, \frac{\partial g(\mathbf{\Gamma}^{(j)})}{\partial \gamma_2}, \ldots, \frac{\partial g(\mathbf{\Gamma}^{(j)})}{\partial \gamma_n} \right]^T$ is the gradient of the response surface g. More details on the numerical quadrature method can be found in the work of Constantine and Gleich [31].

2.3. Computation of the gradient of the response surface $g(\mathbf{\Gamma})$

The computation of the gradient of the response surface $g(\mathbf{\Gamma})$ in Step 4 of Section 2.2 by changing the input parameters of the experimental measurement or numerical simulation is not straightforward, because the computation of the gradient of the response surface includes the determination of π_{ti}' at $\left[\gamma_1{}^{(j)}, \cdots, \gamma_i{}^{(j)} + \Delta h, \cdots, \gamma_n{}^{(j)} \right]$. We must convert the independent quantities $\gamma_i + \Delta h$ for π_{ti}' into the input quantities $\mathbf{q}'$ for the experiment measurement or numerical simulation. We propose the method as follows to calculate the input quantities $\mathbf{q}'$.

The input quantities $\mathbf{q}'$ should meet the condition that only a small Δh increases of γ_i, while γ_j $(i \neq j)$ remains unchanged. Since $\mathbf{x} = \log(\mathbf{q})$, we obtain

$$\mathbf{x}' = \log(\mathbf{q}'). \tag{26}$$

From Eq. (12), we have the relations between the increment of $\mathbf{\Gamma}$ and the input quantities $\mathbf{x}'$ as follows,

$$\begin{aligned} \gamma_i + \Delta h &= \mathbf{w}_i^T \mathbf{x}' \\ \gamma_j &= \mathbf{w}_j^T \mathbf{x}', \end{aligned} \tag{27}$$

where the solution of $\mathbf{x}'$ is not unique and $\mathbf{w}_i$, $\mathbf{w}_j$ are the columns of $\mathbf{W}$ respectively. Because $\mathbf{W}$ is an orthonormal matrix, it is a natural consideration that $\mathbf{x}$ changes along the direction of the base to satisfy Eq. (27). The increment of input parameters of experimental measurements or numerical simulations can be expressed as

Table 1
Dimension vectors of the physical quantities in the pipe flow.

Quantity	Symbol	Unit	Dimension vector
velocity	V	$\mathrm{m\,s^{-1}}$	$[0, 1, -1]$
density	ρ	$\mathrm{kg\,m^{-3}}$	$[1, -3, 0]$
viscosity	μ	$\mathrm{kg\,m^{-1}s^{-1}}$	$[1, -1, -1]$
diameter	D	m	$[0, 1, 0]$
roughness	ε	m	$[0, 1, 0]$

$$\mathbf{x}' = \mathbf{x} + k\mathbf{w}_i. \tag{28}$$

Combining Eq. (27) and Eq. (28), we obtain $k = \Delta h$. The input quantities $\mathbf{q}'$ can be computed as

$$\log(\mathbf{q}') = \log(\mathbf{q}) + \Delta h \mathbf{w}_i. \tag{29}$$

Based on Eq. (29), we can change the input parameters of the experimental measurements or numerical simulations and compute the gradient of the response surface.

3. Machine-learning based data-driven dimensional analysis in the viscous flows in a circular pipe

3.1. Pressure drop in pipe flow

In accordance with the work of Constantine et al. [1], we consider benchmark viscous flows in a circular pipe to validate the machine-learning based data-driven dimensional analysis method. Viscous flow in the pipe is a classical problem involved in dimensional analysis [6]. The physical quantities involved in the system include the average velocity V, density of the fluid ρ, viscosity of the fluid μ, pipe diameter D and pipe wall roughness ε. The system has three fundamental dimensions: length $[L]$, mass $[M]$, and time $[T]$. The dependent variable is the pressure loss dp/dx. The Fanning friction factor λ is defined by

$$\lambda = \frac{dp/dx D}{\frac{1}{2}\rho V^2}. \tag{30}$$

According to the conclusion of classical dimensional analysis and empirical model, the Fanning friction factor λ depends on the Reynolds number $\rho V D/\mu$ and relative roughness ε/D. When the Reynolds number of the pipe is lower than or equal to 3000, it can be considered that the friction factor satisfies the Poiseuille relationship [1]

$$\lambda = \frac{64}{Re}. \tag{31}$$

For the Reynolds number higher than 3000, the Colebrook equation implicitly defines the relationship between the friction factor and the other quantities as follows [1],

$$\frac{1}{\sqrt{\lambda}} = -2.0 \log_{10}\left(\frac{1}{3.7}\frac{\varepsilon}{D} + \frac{2.51}{Re\sqrt{\lambda}}\right). \tag{32}$$

3.2. Machine-learning based data-driven dimensional analysis of flows in a circular pipe

We apply the method proposed in Section 2.2 to determine the dominant dimensionless quantities of viscous flows in a circular pipe. In accordance with the work of Constantine et al. [1], we use a computer model (virtual experiment) of viscous pipe flow to ideally control the system's independent variables and evaluate the corresponding dependent variable. We randomly generate 1000 quantities of virtual experimental data $\left((\lambda^{(1)}, \mathbf{q}^{(1)}), ..., (\lambda^{(1000)}, \mathbf{q}^{(1000)})\right)$. Here, $\mathbf{q} = (\rho, \mu, D, \varepsilon, V)^T$ is the independent quantities with units. The dimension vectors of the independent quantities $\mathbf{q}$ are listed in Table 1.

From the classical dimensional analysis, the equation of the system can be written as

$$\lambda = f\left(Re, \frac{\varepsilon}{D}\right). \tag{33}$$

The matrix $\mathbf{D}$ consists of the dimension vectors list in Table 1. The auxiliary matrix $\mathbf{W}^*$ is computed as the null space of $\mathbf{D}$, and the matrix $\mathbf{W}$ is computed via Schmidt orthogonalization of the auxiliary matrix $\mathbf{W}^*$,

$$\mathbf{D} = \begin{array}{c} \\ M \\ L \\ T \end{array} \begin{array}{c} \rho \quad \mu \quad D \quad \varepsilon \quad V \\ \left[\begin{array}{ccccc} 1 & 1 & 0 & 0 & 0 \\ -3 & -1 & 1 & 1 & 1 \\ 0 & -1 & 0 & 0 & -1 \end{array}\right] \end{array} \tag{34}$$
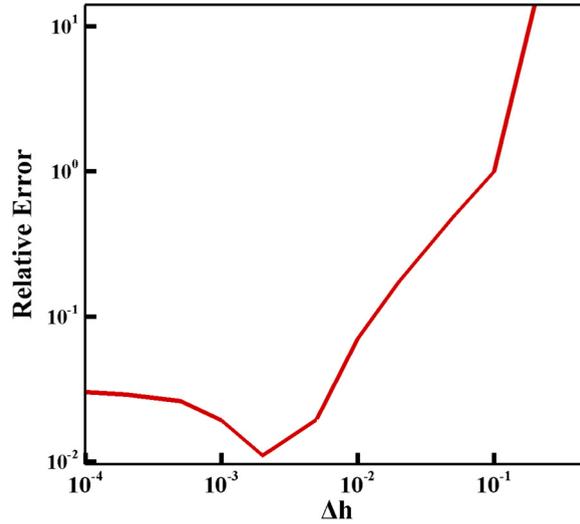
Fig. 2. The relative error (*error*) at different step size Δh.

Table 2
Neural network A.

Hidden layer	Neurons	Activation functions	Optimizer	Batch
5	10	ReLU/Tanh	Adam	20

$$\mathbf{W}^* = \begin{bmatrix} 1 & 0 \\ -1 & 0 \\ 1 & 1 \\ 0 & -1 \\ 1 & 0 \end{bmatrix} \tag{35}$$

$$\mathbf{W} = \begin{bmatrix} 0.5 & \frac{\sqrt{7}}{14} \\ -0.5 & -\frac{\sqrt{7}}{14} \\ 0.5 & -3\frac{\sqrt{7}}{14} \\ 0 & 4\frac{\sqrt{7}}{14} \\ 0.5 & \frac{\sqrt{7}}{14} \end{bmatrix} . \tag{36}$$

We adopt five hidden layers in the neural network in this case. The number of neurons in each hidden layer is ten. The parameters of the neural network are provided in Table 2. The function f_θ in Eq. (19) can be written as

$$\lambda = f_\theta \left(\exp(\mathbf{W}^*) \log(\mathbf{q}) \right). \tag{37}$$

We select 1% samples as the validation set to prevent overfitting. Fig. 3 shows the typical training *loss* and validation *loss* (*val_loss*) of the medium Reynolds case (as discussed in details in the following paragraphs), which are calculated by using Eq. (17). The *loss* of the training set decreases with increasing *epoch*, indicating that the machine learning results have converged. A decrease in *val_loss* indicates that machine learning has increased the prediction ability. With increasing *epoch*, the training *loss* dose not dips significantly below the *val_loss*, which suggests that the neural network does not overfit the sets of data.

We have examined the impact of Δh in Eq. (20) on the results. For the medium Reynolds number case, we alter the step size Δh, and compare the predicted values to the eigenvector calculated with the finite difference-based algorithm. The relative error between the results is defined as

$$error = \frac{||\mathbf{T} - \mathbf{T}'||_2}{||\mathbf{T}'||_2}, \tag{38}$$

where $||.||_2$ is 2-norm, $\mathbf{T}'$ is the theoretical solution and $\mathbf{T}$ is the results with numerical error. Fig. 2 shows the relative error with decreasing Δh. The relative error is closed to 1.0×10^{-2} when $\Delta h < 10^{-2}$. The small error suggests that the results obtained with the proposed method are valid.
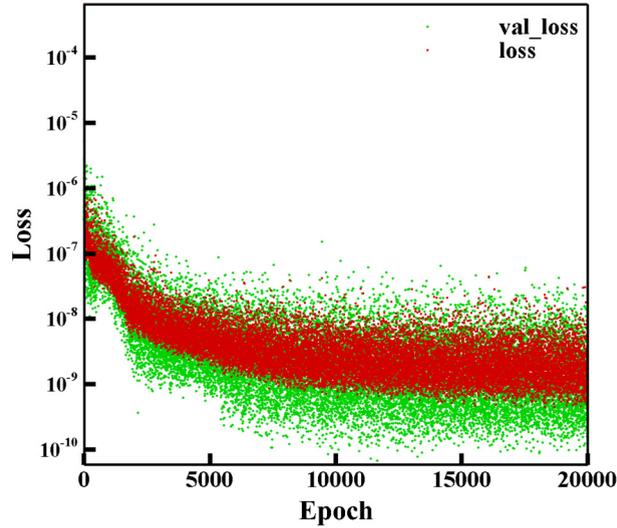
Fig. 3. Changes in *loss* and *val_loss* with *epoch*. The values of both red and green points decrease, *loss* and *val_loss* decrease with *epoch*, which indicates that the result converges and the prediction accuracy increases. The oscillation observed in the figure is related to the small batch size. Increasing the batch size suppresses the resultant oscillation, but local optimal solutions are easily obtained, and *loss* decreases more slowly. Therefore, a small batch size remains the ideal choice for this problem.

Table 3
Bounds of the independent variables in the laminar flow case.

Quantity	Symbol	Lower bound	Upper bound
velocity	V	0.025	0.03
density	ρ	0.1	0.14
viscosity	μ	1×10^{-6}	1×10^{-5}
diameter	D	0.5	0.8
roughness	ε	3×10^{-6}	8×10^{-5}

We first investigate the laminar flows in the pipe. The bounds of the independent variables in the laminar flow case are listed in Table 3. The matrices $\boldsymbol{\Lambda}$, **S** and **Z** of Eq. (13) and Eq. (14) in the laminar flow case are expressed as

$$\boldsymbol{\Lambda} = \begin{bmatrix} 6.95 \times 10^{-1} & \\ & 7.25 \times 10^{-6} \end{bmatrix} \tag{39}$$

$$\mathbf{S} = \begin{bmatrix} 1.00 & 0.00 \\ 0.00 & 1.00 \end{bmatrix} \tag{40}$$

$$\mathbf{Z} = \begin{bmatrix} 0.50 & -0.50 & 0.50 & 0.00 & 0.50 \\ 0.19 & -0.19 & -0.57 & 0.76 & 0.19 \end{bmatrix}. \tag{41}$$

The Fanning friction factor λ can be re-expressed as follows,

$$\begin{aligned} \lambda &= g^*(\hat{\pi}_1, \hat{\pi}_2) \\ &= g^*(\rho^{0.50}\mu^{-0.50}D^{0.50}\varepsilon^{0.00}V^{0.50}, \\ &\quad \rho^{0.19}\mu^{-0.19}D^{-0.57}\varepsilon^{0.76}V^{0.19}). \end{aligned} \tag{42}$$

Based on Eq. (39), we can find that the first eigenvalue is much larger than the second eigenvalue, which indicates that the laminar flow solution is dominated by $\hat{\pi}_1 = \rho^{0.50}\mu^{-0.50}D^{0.50}\varepsilon^{0.00}V^{0.50}$. The result is consistent with that of Constantine et al. [1] and the *Re* number, as shown in Fig. 4. The proposed machine-learning based data-driven dimensional analysis method determines the correct result whereby the Fanning friction factor λ is dominated by the Reynolds number under laminar flow conditions.

We also analyzed the flows in the medium Reynolds number region, as previously analyzed in the work of Constantine et al. [1]. The bounds of the independent variables in the medium Reynolds number case are listed in Table 4. The Reynolds number varies between 1.0×10^4 and 5.6×10^5. In the medium Reynolds number case, the matrices $\boldsymbol{\Lambda}$, **S** and **Z** in Eq. (13) and Eq. (14) are

$$\boldsymbol{\Lambda} = \begin{bmatrix} 4.78 \times 10^{-4} & \\ & 2.86 \times 10^{-5} \end{bmatrix} \tag{43}$$
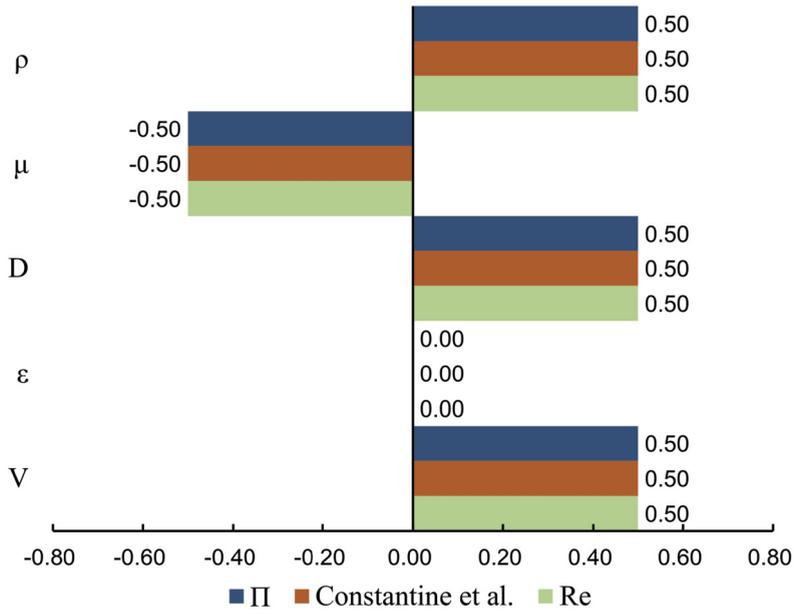
Fig. 4. Comparison of the exponents of dimensionless quantities in the laminar flow case. The green bars indicate the regular Re number. The orange bars indicate the dimensionless quantity of Constantine et al. [1]. And the blue bars indicate the dimensionless quantity of proposed method.

Table 4
Bounds of the independent variables in the medium Reynolds number case.

Quantity	Symbol	Lower bound	Upper bound
velocity	V	2.0	4.0
density	ρ	0.1	0.14
viscosity	μ	1×10^{-6}	1×10^{-5}
diameter	D	0.5	1.0
roughness	ε	5×10^{-4}	2×10^{-3}

$$\mathbf{S} = \begin{bmatrix} 0.84 & 0.55 \\ -0.55 & 0.84 \end{bmatrix} \tag{44}$$

$$\mathbf{Z} = \begin{bmatrix} 0.31 & -0.31 & 0.73 & -0.41 & 0.31 \\ 0.43 & -0.43 & -0.20 & 0.63 & 0.43 \end{bmatrix}. \tag{45}$$

The Fanning friction factor λ can be re-expressed as follows,

$$\begin{aligned} \lambda =& g^*(\hat{\pi}_1, \hat{\pi}_2) \\ =& g^*(\rho^{0.31}\mu^{-0.31}D^{0.73}\varepsilon^{-0.41}V^{0.31}, \\ & \rho^{0.43}\mu^{-0.43}D^{-0.20}\varepsilon^{0.63}V^{0.43}). \end{aligned} \tag{46}$$

Equation (43) indicates that the two eigenvalues are of the same order. The Fanning friction factor λ depends on both $\hat{\pi}_1$ and $\hat{\pi}_2$. The results indicate that the Fanning friction factor λ depends on two dimensionless quantities in the Reynolds number region $(1.0 \times 10^4 < Re < 5.6 \times 10^5)$.

The dimensionless quantities obtained by the method proposed by this work and these reported in the work of Constantine et al. [1] are shown in Fig. 5. These two dimensionless quantities are close with those obtained by Constantine et al. [1] with difference between 0.01 to 0.02. The cause of the difference might be that the number of points Constantine et al. [1] considered in their data-driven dimensional analysis algorithm was 161051, and we can obtain the similar result by using the neural network with 1000 random points.

In the high Reynolds number case, Re varies between 2.5×10^6 and 9.8×10^7. The bounds of the independent variables are listed in Table 5. In the high Reynolds number case, the matrices $\mathbf{\Lambda}$, $\mathbf{S}$ and $\mathbf{Z}$ are expressed as

$$\mathbf{\Lambda} = \begin{bmatrix} 2.68 \times 10^{-6} & \\ & 7.69 \times 10^{-3} \end{bmatrix} \tag{47}$$

$$\mathbf{S} = \begin{bmatrix} -0.93 & 0.36 \\ -0.36 & -0.93 \end{bmatrix} \tag{48}$$
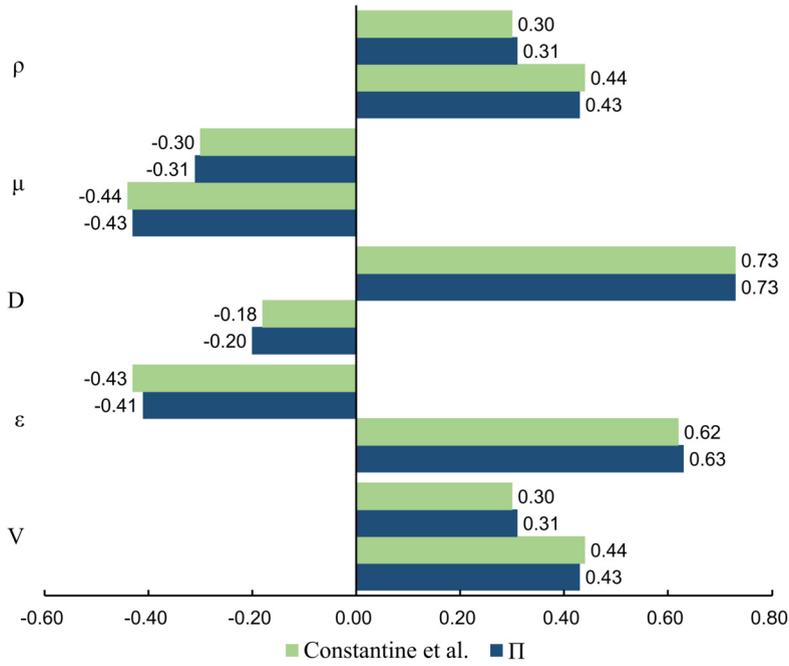
Fig. 5. Comparison of the exponents of dimensionless quantities in the medium Reynolds number case. The green bars indicate the dimensionless quantities of Constantine et al. [1]. And the blue bars indicate the dimensionless quantities of proposed method. The two dimensionless quantities are equally important, they are both presented in the figure.

Table 5
Bounds on the independent variables for the high Reynolds number case.

Quantity	Symbol	Lower bound	Upper bound
velocity	V	500	700
density	ρ	0.1	0.14
viscosity	μ	1×10^{-6}	1×10^{-5}
diameter	D	0.5	1.0
roughness	ε	1×10^{-2}	4×10^{-2}

$$\mathbf{Z} = \begin{bmatrix} -0.53 & 0.53 & -0.26 & -0.27 & -0.53 \\ 0.00 & 0.00 & 0.71 & -0.71 & 0.00 \end{bmatrix}. \tag{49}$$

The Fanning friction factor λ can be rewritten as follows,

$$\begin{aligned} \lambda &= g^*(\hat{\pi}_1, \hat{\pi}_2) \\ &= g^*(\rho^{-0.53}\mu^{0.53}D^{-0.26}\varepsilon^{-0.27}V^{-0.53}, \\ &\quad \rho^{0.00}\mu^{0.00}D^{0.71}\varepsilon^{-0.71}V^{0.00}). \end{aligned} \tag{50}$$

Equation (47) reveals that the second eigenvalue is much larger than the first eigenvalue, which suggests that the solution under high Reynolds number flow conditions is dominated by $\hat{\pi}_2$, the pipe wall roughness, as shown in Fig. 6.

We also present an example with real experimental data (Nikuradse's data) available from the work of Yang et al. [32] for the high Reynolds number case. Yang et al. [32] describes the collection for Nikuradse's data with six different roughnesses. We choose the cases in which the Reynolds number Re varies between 1.0×10^5 and 2.0×10^6. The max relative error of data is 4.03% and the average relative error is 1.01%. By using these data with errors, the proposed method based on the neural network gives the matrices $\mathbf{\Lambda}$, $\mathbf{S}$ and $\mathbf{Z}$ as follows,

$$\mathbf{\Lambda} = \begin{bmatrix} 9.30 \times 10^{-5} & \\ & 4.68 \times 10^{-4} \end{bmatrix} \tag{51}$$

$$\mathbf{S} = \begin{bmatrix} -0.90 & 0.44 \\ -0.44 & -0.90 \end{bmatrix} \tag{52}$$

$$\mathbf{Z} = \begin{bmatrix} -0.53 & 0.53 & -0.20 & -0.44 & -0.53 \\ 0.05 & -0.05 & 0.73 & -0.68 & 0.05 \end{bmatrix}. \tag{53}$$
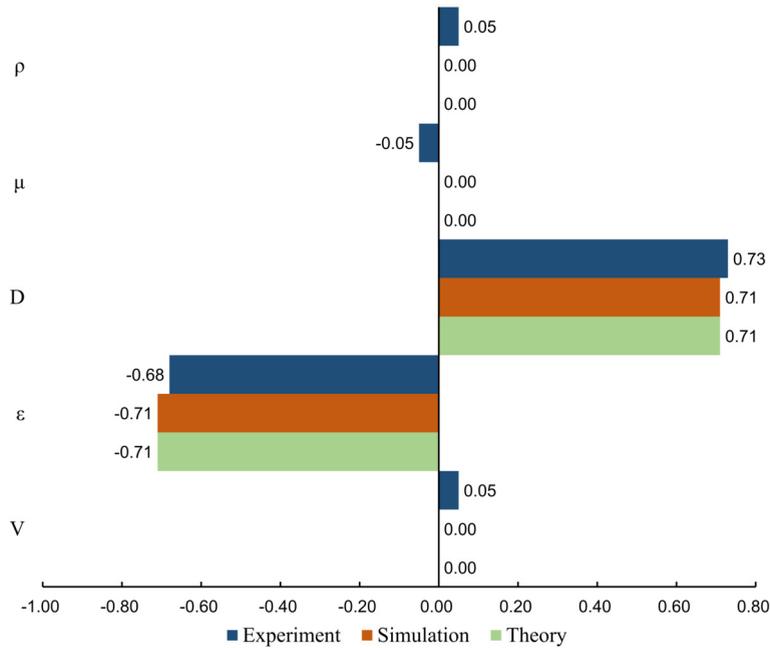
Fig. 6. Comparison of the exponents of dimensionless quantities in the high Reynolds number case. The green bars indicate the D/ε. The orange bars indicate the dimensionless quantity of virtual simulation applying proposed method. And the blue bars indicate the dimensionless quantity of experimental data [32] applying proposed method.

Compared with Eq. (49), which is the results of virtual experiment, the exponents of V, ρ and μ in Eq. (53) are 0.05 rather than the theoretical results which are 0.00. And the exponents of D in Eq. (53) is 0.73 and the exponents of ε is -0.68. The ratio of eigenvalues shown in Eq. (51) is 5.03. This result can still show the non-dimensional quantity corresponds to the second eigenvector D/ε is the relevant quantity at high Re, when the number is round to the nearest tenth. However, we would like to notice that the error in the experimental data has great effects on the ratio between the two eigenvalues. Much attention should be paid when the proposed method is used to handle the data with errors, since the ratio between the eigenvalues of shown in Eq. (51) is not as large as that shown in Eq. (47) for the virtual experiments.

3.3. Discussion

The numerical errors in the proposed method based on the neural network mainly origin from the three steps as follows: training the neural network for response surface (Step 3 in Section 2.2), approximating the gradients of the response surface (Step 4 in Section 2.2), evaluating the integral by using numerical quadrature (Step 5 in Section 2.2). We investigate the propagation of the error from Step 3 to Step 5 based on the case of pipe flow in medium Reynolds number.

We first calculated the propagation of numerical errors for the case at medium Reynolds numbers as reported in Section 3.2, where 1000 sets of data are used. The errors in Steps $3-5$ are computed based on Eq. (38) according to the rules as follows: (1) the error in Step 3 is defined based on the difference between the predicted of the artificial neural network and the theoretical results, (2) the error in Step 4 is defined based on the difference between the gradient of the finite difference method and the gradient of Colebrook function or Poiseuille function at the same point, (3) the error in Step 5 is defined based on the difference between the numerical quadrature and the integration of continuous function.

For the case with 1000 sets of data shown in Fig. 7, the error of training the neural network for response surface (Step 3 in Section 2.2) is 0.73%, while the error after approximating the gradients of the response surface (Step 4 in Section 2.2) is 4.47%. It is reasonable that the error is amplified during the computation of gradient, since error from the approximation of the response surface is divided by a small increment Δh. The error after evaluating the integral by using numerical quadrature (Step 5 in Section 2.2) is 4.87%, which is closed to error of Step 4.

We have also changed the number of data sets while keep all of the other parameters fixed. The number of data sets investigated in this subsection varies from 200 to 1500. The propagation of the errors has the similar trends as these reported for the case with 1000 sets of data. However, the errors after evaluating the integral by using numerical quadrature (Step 5 in Section 2.2) are more than 10% for the cases with 200 and 500 sets of data. For example, the error after evaluating the integral by using numerical quadrature (Step 5 in Section 2.2) is 10.94% for the case with 500 sets of data. When the number of data set is larger than 1000, the error after evaluating the integral by using numerical quadrature (Step 5 in Section 2.2) is less than 5%.

The main differences between the proposed neural network based method and the previous work of Constantine et al. [1] are the approaches to compute the gradient of the ridge function. Constantine et al. [1] present two different approaches
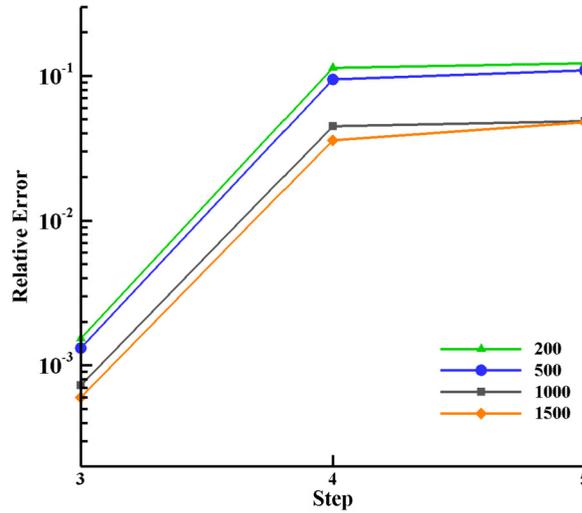
Fig. 7. The propagation of the numerical error from Steps 3–5 reported in Section 2.2.

Table 6
The relative errors of different sets of data.

Method	Sets of data	Relative error
proposed method	1000	4.87%
proposed method	10000	3.09%
Gaussian process regression	1000	54.70%
Gaussian process regression	2000	21.28%
Gaussian process regression	4000	13.69%
Gaussian process regression	10000	11.58%
Gaussian process regression	16000	5.03%

for numerically computing the gradients of the ridge function. The first approach is based on the response surface where the gradient of response surface is used to approximate the gradient of the ridge function. The second approach is based on the finite difference approximations of the ridge function. The finite-difference based approach does not surfer from the error from response surface approximation. However, it requires more experiments than the response-surface based method [1]. Constantine et al. [1] use Gaussian process regression to construction the response surface in analyzing the viscous pipe flow. Here, we compared the proposed neural network based method with the Gaussian process regression based method. We investigate the errors in analyzing the viscous pipe flow at medium Reynolds numbers as reported in Section 3.2. We use the same ways to compute the gradient of the response surface and the numerical quadrature in Eq. (38). The only differences are the way to construct the response surface and the number of data sets.

We use the open source machine learning library, _Scikit-learn_, to implement the Gaussian process regression by calling the function _sklearn.gaussian_process.GaussianProcessRegressor_. For the analysis based on the 1000 sets of data reported in Section 3.2 at medium _Re_ number, the error of matrix **C** computed by using the proposed neural network based method is 4.87%, while that computed by using the Gaussian process regression based method is more than 20%. The error can be reduced if we increase the number of data sets for constructing the response surface, as shown in Table 6. The results show that the proposed neural network based method requires less sets of data than the Gaussian process regression based method. We would like to notice that it is hard to clearly clarify the advantage of the proposed neural network based method, because we just use one of the Gaussian process regression method from the library _Scikit − learn_. However, the result at least shows that the proposed neural network based method provides an alternative way to construct the response surface for data-driven dimensional analysis.

It is noticed that the proposed method is a data-driven method without referencing to the physical models in analyzing the data. Compared with the physical inferred model, the data-driven method has the drawback of requiring massive sets of data, which might be quite expensive in terms of scientific computing. Constantine et al. [1] uses 161051 sets of data in their analyzing of the pipe flows. Through the present work based on the neural network tries to reduce the required sets of data, it still requires 1000 sets of data to obtain the acceptable results. A combination of the physical inferred model with the data-driven model might be a feasible way to reduce the required data sets. As will be discussed in Section 4, the physical inferred assumption that the drag coefficient is independent of dimensionless soap film thickness will significantly reduce the required sets of data. For the analysis of drag coefficients of a flexible body, the proposed method requires just about 100 sets of data with the above physical inferred assumption, while that without the physical inferred assumption requires about 1000 sets of data. The other disadvantage of the neural network based method is that experience is needed

in setting and training neural networks to avoid underfitting and overfitting. Attentions should be paid in determining the number of hidden layers and neurons.

We also would like to mention that the current data-driven dimensional analysis is constructed based on the assumption as follows: the data available are sufficient to discover all of the related physical quantities and there are no hidden or unavailable inputs. In general, we assume that all of the related physical quantities have been considered. In practical applications, we need to consider as many independent variables as possible, even those seemingly unrelated variables. If important variables are lurking, they can be detected via the method proposed by Rosario et al. [33].

4. Machine-learning based data-driven dimensional analysis of drag coefficients of a flexible body

4.1. Interactions between a flexible body and flows

Reconfiguration of a flexible body in flows is a common phenomenon in nature. Biological structures are found to fold, bend, wave, and twist in flows [34]. Reconfiguration of a given flexible body yields different scaling of drag coefficient. Vogel et al. [35] studied the relationship between drag and shape reconfiguration of tree leaves at different wind speeds. With increasing wind speed, the leaves roll themselves into self-similar cones and tend to become streamlined. The self-similar feature of a flexible body results in a slower drag growth than that of a rigid body.

Scaling of the drag coefficient of a flexible body in flows was systematically investigated by experimental measurements [36,37] and numerical simulations [38–40]. Alben et al. [36,37] found that the drag coefficient scales as $U^{4/3}$ for flexible body instead of as U^2 for rigid body.

The new scaling law of the drag acting a flexible body is obtained by proposing a crucial dimensionless length η to analyze the drag coefficient [36]. The dimensionless length η was obtained by analyzing the coupled equations of the free-streamline model and the Euler-Bernoulli beam equation as

$$-(T\mathbf{s})_s + (Ek_s\mathbf{n})_s = hp\mathbf{n}, \tag{54}$$

where p is the pressure jump across the fiber, T is the axial tension, k is the fiber curvature, $\mathbf{n}$ is the unit normal vector, subscript s is the arc length and $\mathbf{s}$ is the unit tangent vector along the fiber. The dimensionless form of Eq. (54) is

$$k_{ss} + k^3/2 = \eta^2 p, \tag{55}$$

where the dimensionless length η is expressed as

$$\eta = \left(\frac{\rho h l^2 V^2/2}{E/l} \right)^{\frac{1}{2}}. \tag{56}$$

4.2. Dominant dimensionless quantities of the fluid-structure interaction system

In this section, we show that the crucial dimensionless length η can be obtained by using the machine-learning based data-driven dimensional analysis proposed in this work without referring to the complex coupled equations of the free-streamline model and the Euler-Bernoulli beam equation. The *Drag* acting on the fiber depends on the length of the flexible fiber l, elastic modulus of the flexible fiber E, fluid density ρ, thickness of the soap film h, flow velocity V, and viscosity coefficient of the fluid μ. Then, we obtain the expression as follows,

$$Drag = f(l, E, \rho, h, V, \mu). \tag{57}$$

We can obtain three dimensionless quantities according to the classical dimensional analysis method, namely, $\rho l V/\mu$, $\rho l^4 V^2/E$, and h/l. Equation (57) can be rewritten with dimensionless quantities as follows,

$$C_d = f\left(\frac{\rho l V}{\mu}, \frac{\rho l^4 V^2}{E}, \frac{h}{l} \right), \tag{58}$$

where $C_d = Drag/\left(\rho U^2 lh/2\right)$ is the drag coefficient.

We apply the machine-learning based data-driven dimensional analysis method given in Section 2.2 to investigate the above fluid-structure interaction system. We generate 1000 quantities of virtual experimental data $\left(\left(Cd^{(1)}, \mathbf{q}^{(1)}\right), ..., \left(Cd^{(1000)}, \mathbf{q}^{(1000)}\right) \right)$ according to the relationship with a random noise in a similar approach reported in reference [37]. Here, $\mathbf{q} = (\rho, \mu, l, h, V, E)^T$ are the dimensional quantities (independent variables) in the system. The bounds of the quantities $\mathbf{q}$ are listed in Table 7.
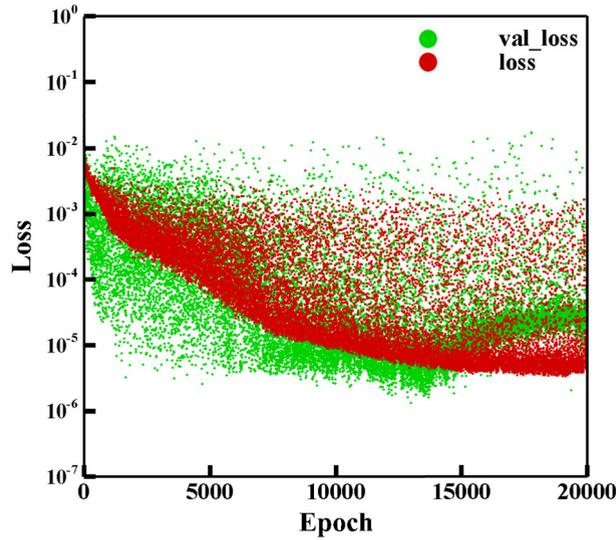
Fig. 8. Changes in *loss* and *val_loss* with *epoch*. The values of both red and green points before *epoch* reach 15000 decrease. After *epoch* reaches 15000, the values of the green points increase, *val_loss* increases, the neural network experiences a risk of overfitting, and *loss* does not considerably decrease. We chose 15000 *epoch* in the simulations.

Table 7
Bounds of the independent variables in the flexible body case.

Physical quantity	Symbol	Lower bound	Up bound
fluid density	ρ	0.8	1.2
fluid velocity	V	0.0	5.0
fiber length	l	4.0	8.0
film thickness	h	0.1	0.4
fiber rigidity	E	1.0	4.0

We can obtain matrices **D**, **W*** and **W** of this problem as follows

$$\mathbf{D} = \begin{array}{c} M \\ L \\ T \end{array} \begin{array}{c} \begin{array}{cccccc} \rho & \mu & l & h & V & E \end{array} \\ \left[\begin{array}{cccccc} 1 & 1 & 0 & 0 & 0 & 1 \\ -3 & -1 & 1 & 1 & 1 & 3 \\ 0 & -1 & 0 & 0 & -1 & -2 \end{array} \right] \end{array} \tag{59}$$

$$\mathbf{W}^* = \left[\begin{array}{ccc} 0 & 1 & 1 \\ 0 & -1 & 0 \\ 1 & 1 & 4 \\ -1 & 0 & 0 \\ 0 & 1 & 2 \\ 0 & 0 & -1 \end{array} \right] \tag{60}$$

$$\mathbf{W} = \left[\begin{array}{ccc} 0 & \sqrt{14}/7 & -\sqrt{21}/28 \\ 0 & -\sqrt{14}/7 & 5\sqrt{21}/42 \\ \sqrt{2}/2 & \sqrt{14}/14 & 3\sqrt{21}/28 \\ -\sqrt{2}/2 & \sqrt{14}/14 & 3\sqrt{21}/28 \\ 0 & \sqrt{14}/7 & \sqrt{21}/21 \\ 0 & 0 & -\sqrt{21}/12 \end{array} \right]. \tag{61}$$

The hidden layers of the neural network are the same as the first case in Table 2.

$$C_d = f_\theta \left(\exp(\mathbf{W}^*) \log(\mathbf{q}) \right) \tag{62}$$

Fig. 8 shows the training *loss* and the validation *loss* (*val_loss*) of the flexible body case. In this case, *val_loss* exhibits a downward trend before *epoch* reaches 15000, and *val_loss* reveals an upward trend after *epoch* reaches 15000, while *loss* remains almost unchanged, as shown in Fig. 8. The accuracy of the neural network does not further increase after *epoch*

Table 8
Comparison of the exponents of quantities π_η and $\hat{\pi}_1$.

Dimensionless quantity	ρ	μ	l	h	V	E
$\hat{\pi}_1$	−0.25	0.00	−0.75	−0.25	−0.50	0.25
π_η	−0.25	0.00	−0.75	−0.25	−0.50	0.25

reaches 15000. Therefore, we choose 15000 *epoch*. Following the steps of the proposed method reported in Section 2.2, the matrices $\mathbf{\Lambda}$, $\mathbf{S}$ and $\mathbf{Z}$ are expressed as

$$\mathbf{\Lambda} = \begin{bmatrix} 3.38 \times 10^0 & & \\ & 4.10 \times 10^{-3} & \\ & & 8.54 \times 10^{-4} \end{bmatrix} \tag{63}$$

$$\mathbf{S} = \begin{bmatrix} -0.36 & -0.88 & -0.31 \\ -0.67 & 0.47 & -0.58 \\ -0.65 & -0.01 & 0.76 \end{bmatrix} \tag{64}$$

$$\mathbf{Z} = \begin{bmatrix} -0.25 & 0.00 & -0.75 & -0.25 & -0.50 & 0.25 \\ 0.25 & -0.25 & -0.50 & 0.75 & 0.25 & 0.00 \\ -0.43 & 0.72 & -0.00 & 0.44 & -0.14 & -0.29 \end{bmatrix}. \tag{65}$$

We obtain the new dimensionless quantities as follows,

$$\begin{aligned} C_d &= g^*(\hat{\pi}_1, \hat{\pi}_2, \hat{\pi}_3) \\ &= g^*(\rho^{-0.25} \mu^{0.00} l^{-0.75} h^{-0.25} V^{-0.50} E^{0.25}, \\ &\quad \rho^{-0.25} \mu^{-0.25} l^{-0.50} h^{0.75} V^{0.25} E^{0.00}, \\ &\quad \rho^{-0.43} \mu^{0.72} l^{-0.00} h^{0.44} V^{-0.14} E^{-0.29}). \end{aligned} \tag{66}$$

According to the definition of the active subspace, the dimensionless quantities are evaluated according to the eigenvalues. Based on the results, it is observed that the first eigenvalue in Eq. (63) is much larger than the other two eigenvalues. Hence, the first eigenvector is the dominant direction,

$$\hat{\pi}_1 = \rho^{-0.25} \mu^{0.00} l^{-0.75} h^{-0.25} V^{-0.50} E^{0.25}, \tag{67}$$

which is the dominant dimensionless quantity of the fluid-structure interaction system, whereas the other two dimensionless quantities impose little effect on the drag coefficient. Eq. (56) reveals that η can be expressed as

$$\eta = (\rho^{-0.25} \mu^{0.00} l^{-0.75} h^{-0.25} V^{-0.50} E^{0.25})^{-2} / \sqrt{2}. \tag{68}$$

A comparison of the dimensionless quantity $\hat{\pi}_1$ of the active subspace and the regularized exponents term of η, which is in parentheses in Eq. (68), π_η is presented in Table 8. This dimensionless quantity $\hat{\pi}_1$ is equivalent to η mentioned in Section 4.1.

The drag coefficient of a flexible body is closely related to η and is slightly related to the other dimensionless quantities. This result is consistent with the conclusion in the work of Alben [36].

It is shown that the machine-learning based data-driven dimensional analysis can reveal the dominant dimensionless quantities by only considering relevant data without need to refer to the free-streamline model and Euler-Bernoulli beam equations in Eq. (54). The dominant dimensionless quantity η derived via machine learning is consistent with the results obtained via the theoretical analysis. These results demonstrate that machine-learning based data-driven dimensional analysis method can help us identify and extract the dominant dimensionless quantities of physical systems.

Furthermore, we know that the drag coefficient is related to η, and in experiments [36], the results were independent of dimensionless quantity (dimensionless soap film thickness) h/l. Hence, due to the reduction in variables, the problem of flexible body drag reduction is only related to two dimensionless quantities, and the other dimensionless quantity is Re. Therefore, we can change Eq. (62) as

$$C_d = f_\theta \left(\exp(\mathbf{W}') \log(\mathbf{q}) \right), \mathbf{W}' = \begin{bmatrix} 1 & 1 \\ -1 & 0 \\ 1 & 3 \\ 0 & 1 \\ 1 & 2 \\ 0 & -1 \end{bmatrix}^T. \tag{69}$$

Under these settings, with only a small amount of experimental data, we can obtain valid conclusions.

Fig. 9. Comparison of the exponents of dimensionless quantities in flexible body case. The green bars indicate the dimensionless quantity obtained with theory. The orange bars indicate the dimensionless quantity of virtual simulation. And the blue bars indicate the dimensionless quantity of experimental results.

Table 9
Neural network B.

Hidden layer	Neurons	Activation functions	Optimizer	Batch
2	4	ReLU/Tanh	Adam	2

We report the application of the proposed method to the experimental data reported in [36] when η is larger than 1. Due to the small amount of data, the structure of the neural network is summarized in Table 9.

Following the same steps reported in Section 2.2, we obtain

$$\mathbf{\Lambda} = \begin{bmatrix} 8.67 \times 10^1 & & \\ & 3.72 \times 10^{-1} & \\ & & 1.33 \times 10^{-1} \end{bmatrix} \tag{70}$$

$$\mathbf{S} = \begin{bmatrix} -0.37 & -0.88 & 0.29 \\ -0.66 & 0.02 & -0.75 \\ -0.66 & 0.47 & 0.59 \end{bmatrix} \tag{71}$$

$$\mathbf{Z} = \begin{bmatrix} -0.24 & -0.01 & -0.76 & -0.24 & -0.50 & 0.25 \\ -0.06 & 0.24 & -0.38 & 0.86 & 0.12 & -0.18 \\ -0.50 & 0.72 & 0.29 & -0.12 & -0.28 & -0.22 \end{bmatrix}. \tag{72}$$

According to the proposed dimensional analysis, we write the results of the dimensionless quantities as follows,

$$\begin{aligned} C_d = g^*(&\rho^{-0.24}\mu^{-0.01}l^{-0.76}h^{-0.24}V^{-0.50}E^{0.25}, \\ &\rho^{-0.06}\mu^{0.24}l^{-0.38}h^{0.86}V^{0.12}E^{-0.18}, \\ &\rho^{-0.50}\mu^{0.72}l^{0.32}h^{-0.12}V^{-0.28}E^{-0.22}). \end{aligned} \tag{73}$$

The first eigenvalue in Eq. (70) is much larger than the other eigenvalues, so the drag coefficient is highly correlated with $\hat{\pi}_1$. The results obtained via analysis of the real experimental data are consistent with those obtained via Eq. (67). Under the premise of determining the dimensionless quantities, this demonstrates that by analyzing a small amount of experimental data, the method can also be applied to determine the *unique* and *relevant* dimensionless quantities. In conclusion, the exponents of dimensionless quantities obtained with proposed method and theory are shown in Fig. 9.

5. Conclusion

The two major limitations of the classical dimensional analysis method are that (1) the dimensionless quantities are not unique, and (2) the relative importance of the different dimensionless quantities is not measured. We proposed a machine-learning based data-driven dimensional analysis method to address the two major limitations. The proposed method identifies *unique* and *relevant* dimensionless quantities based on the data-driven dimensional analysis proposed by Constantine et al. [1], which combines the classical dimensional analysis with the active subspace. The active subspace is estimated from the ridge function corresponding to the response surface. We use a fully connected neural network to construct the response surface and provide an effective approach to determine the increment of input quantities in order to compute the gradient of the response surface. The proposed method is validated by using the simulations and experiments for benchmark viscous pipe flows and a flexible body interacting with a uniform upstream flows. In the benchmark cases of flows in pipes, the dominant dimensionless quantities at different regions of Reynolds number have been successfully identified. The proposed method has obtained accurate results from a relatively small amount of data. In the fluid-structure interaction problem involving drag reduction through the self-similar bending of a flexible body, the proposed method has obtained the unique and dominant dimensionless quantity without referring to the complex equations involving the fluid dynamics and structure dynamics.

CRediT authorship contribution statement

Zhaoyue Xu: Data curation, Formal analysis, Methodology, Writing – original draft. **Xinlei Zhang:** Formal analysis, Methodology. **Shizhao Wang:** Conceptualization, Formal analysis, Writing – review & editing. **Guowei He:** Conceptualization, Formal analysis, Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

References

[1] P.G. Constantine, Z. del Rosario, G. Iaccarino, Data-driven dimensional analysis: algorithms for unique and relevant dimensionless groups, arXiv e-prints, arXiv:1708.04303, 2017.

[2] V. Simon, B. Weigand, H. Gomaa, Dimensional Analysis for Engineers, Springer, 2017.

[3] A.C. Palmer, Dimensional Analysis and Intelligent Experimentation, World Scientific, 2008.

[4] L. Jofre, Z.R. del Rosario, G. Iaccarino, Data-driven dimensional analysis of heat transfer in irradiated particle-laden turbulent flow, Int. J. Multiph. Flow 125 (2020) 10319.

[5] L.L. Henry, Dimensional Analysis and Theory of Models, Franklin Inst., 1951.

[6] Q. Tan, Dimensional Analysis with Case Studies in Mechanics, Springer-Verlag Berlin Heidelberg, 2011.

[7] P.G. Constantine, Active Subspaces: Emerging Ideas for Dimension Reduction in Parameter Studies, SIAM, 2015.

[8] P.G. Constantine, E. Dow, Q. Wang, Active subspace methods in theory and practice: applications to kriging surfaces, SIAM J. Sci. Comput. 36 (2014) A1500–A1524.

[9] H.M. Raymond, Response surface methodology—current status and future directions, J. Qual. Technol. 31 (1999) 30–44.

[10] A.I. Khuri, S. Mukhopadhyay, Response surface methodology, Wiley Interdiscip. Rev.: Comput. Stat. 2 (2010) 128–149.

[11] X. Yang, H. Lei, N.A. Baker, G. Lin, Enhancing sparsity of Hermite polynomial expansions by iterative rotations, J. Comput. Phys. 307 (2016) 94–109.

[12] K. Fukumizu, C. Leng, Gradient-based kernel dimension reduction for regression, J. Am. Stat. Assoc. 109 (2014) 359–370.

[13] A. Glaws, P.G. Constantine, R. Cook, Inverse regression for ridge recovery: a data-driven approach for parameter reduction in computer experiments, Stat. Comput. 30 (2020) 237–253.

[14] Z. Deng, C. He, Y. Liu, Deep neural network-based strategy for optimal sensor placement in data assimilation of turbulent flow, Phys. Fluids 33 (2021) 025119.

[15] Y. Yin, P. Yang, Y. Zhang, H. Chen, S. Fu, Feature selection and processing of turbulence modeling based on an artificial neural network, Phys. Fluids 32 (2020) 105117.

[16] X. Zhang, J. Wu, O. Coutier-Delgosha, H. Xiao, Recent progress in augmenting turbulence models with physics-informed machine learning, J. Hydrodyn. 31 (2019) 1153–1158.

[17] C.A.M. Ströfer, X. Zhang, H. Xiao, Ensemble gradient for learning turbulence models from indirect observations, arXiv preprint, arXiv:2104.07811, 2021.

[18] C. Xie, Z. Yuan, J. Wang, Artificial neural network-based nonlinear algebraic models for large eddy simulation of turbulence, Phys. Fluids 32 (2020) 115101.

[19] X. Yang, S. Zafar, J. Wang, H. Xiao, Predictive large-eddy-simulation wall modeling via physics-informed neural networks, Phys. Rev. Fluids 4 (2019) 034602.

[20] B. Han, W. Huang, Active control for drag reduction of turbulent channel flow based on convolutional neural networks, Phys. Fluids 32 (2020) 095108.

[21] H. Wang, Z. Yang, B. Li, S. Wang, Predicting the near-wall velocity of wall turbulence using a neural network for particle image velocimetry, Phys. Fluids 32 (2020) 115105.

[22] M. Xu, S. Song, X. Sun, W. Zhang, A convolutional strategy on unstructured mesh for the adjoint vector modeling, Phys. Fluids 33 (2021) 036115.

[23] M. Farooq Anjum, I. Tasadduq, K. Al-Sultan, Response surface methodology: a neural network approach, Eur. J. Oper. Res. 101 (1997) 65–73.

[24] S. Sada, Modeling performance of response surface methodology and artificial neural network, J. Appl. Sci. Environ. Manag. 22 (2018) 875.

[25] C.F. Matta, L. Massa, A.V. Gubskaya, E. Knoll, Can one take the logarithm or the sine of a dimensioned quantity or a unit? Dimensional analysis involving transcendental functions, J. Chem. Educ. 88 (2011) 67–70.

[26] P. Molyneux, The dimensions of logarithmic quantities: implications for the hidden concentration and pressure units in ph values, acidity constants, standard thermodynamic functions, and standard electrode potentials, J. Chem. Educ. 68 (1991) 467.

[27] A. Pinkus, Ridge Function, Cambridge University Press, 2015.

[28] J.M. Hokanson, P.G. Constantine, Data-driven polynomial ridge approximation using variable projection, SIAM J. Sci. Comput. 40 (2018) A1566–A1589.

[29] Y. Bengio, I. Goodfellow, A. Courville, Deep Learning, MIT Press, 2016.

[30] D.P. Kingma, J.L. Ba, Adam: a method for stochastic optimization, in: International Conference on Learning Representations, ICLR, 2015.

[31] P.G. Constantine, D.F. Gleich, Computing active subspaces with Monte Carlo, arXiv e-prints **math**, arXiv:1408.0545, 2015.

[32] B.H. Yang, D.D. Joseph, Virtual Nikuradse, J. Turbul. 10 (2009) N11.

[33] Z. del Rosario, M. Lee, G. Iaccarino, Lurking variable detection via dimensional analysis, Int. J. Uncertain. Quantificat. 7 (2019) 232–259.

[34] M. Shelley, J. Zhang, Flapping and bending bodies interacting with fluid flows, Annu. Rev. Fluid Mech. 43 (2011) 449–465.

[35] S. Vogel, Drag and reconfiguration of broad leaves in high winds, J. Exp. Bot. 40 (1989) 941–948.

[36] S. Alben, M. Shelley, J. Zhang, Drag reduction through self-similar bending of a flexible body, Nature 420 (2003) 479–481.

[37] S. Alben, M. Shelley, J. Zhang, How flexibility induces streamlining in a two-dimensional flow, Phys. Fluids 16 (2004) 1694–1713.

[38] L. Zhu, C. Peskin, Drag of a flexible fibre in a 2D moving viscous fluid, Comput. Fluids 36 (2007) 398–406.

[39] L. Zhu, X. Yu, N. Liu, Y. Cheng, X. Lu, A deformable plate interacting with a non-Newtonian fluid in three dimensions, Phys. Fluids 29 (2017) 083101.

[40] L. Zhu, G. He, S. Wang, L. Miller, X. Zhang, Q. You, S. Fang, An immersed boundary method based on the lattice Boltzmann approach in three dimensions, with application, Comput. Math. Appl. 61 (2011) 3506–3518.