

Article

# Plane Cascade Aerodynamic Performance Prediction Based on Metric Learning for Multi-Output Gaussian Process Regression

Lin Liu <sup>1</sup> , Chunming Yang <sup>1,\*</sup> , Honghui Xiang <sup>2</sup> and Jiazhe Lin <sup>3</sup>

<sup>1</sup> School of Computer Science and Technology, Southwest University of Science and Technology, Mianyang 621010, China; ll666ll@mails.swust.edu.cn

<sup>2</sup> AECC Sichuan Gas Turbine Establishment, Mianyang 621000, China; xianghonghui624@sina.com

<sup>3</sup> China Aerodynamics Research and Development Center, Computational Aerodynamic Research Institute, Mianyang 621000, China; linjiazhe@cardc.cn

\* Correspondence: yangchunming@swust.edu.cn

**Abstract:** Multi-output Gaussian process regression measures the similarity between samples based on Euclidean distance and assigns the same weight to each feature. However, there are significant differences in the aerodynamic performance of plane cascades composed of symmetric and asymmetric blade shapes, and there are also significant differences between the geometry of the plane cascades formed by different blade shapes and the experimental working conditions. There are large differences in geometric and working condition parameters in the features, which makes it difficult to accurately measure the similarity between different samples when there are fewer samples. For this problem, a metric learning for the multi-output Gaussian process regression method (ML\_MOGPR) for aerodynamic performance prediction of the plane cascade is proposed. It shares parameters between multiple output Gaussian distributions during training and measures the similarity between input samples in a new embedding space to reduce bias and improve overall prediction accuracy. For the analysis of ML\_MOGPR prediction results, the overall prediction accuracy is significantly improved compared with multi-output Gaussian process regression (MOGPR), backpropagation neural network (BPNN), and multi-task learning neural network (MTLNN). The experimental results show that ML\_MOGPR is effective in predicting the performance of the plane cascade, and it can quickly and accurately make a preliminary estimate of the aerodynamic performance and meet the performance parameter estimation accuracy requirements in the early stage.

**Keywords:** machine learning; aerodynamic analysis; multiple outputs; Gaussian process regression; metric learning



**Citation:** Liu, L.; Yang, C.; Xiang, H.; Lin, J. Plane Cascade Aerodynamic Performance Prediction Based on Metric Learning for Multi-Output Gaussian Process Regression. *Symmetry* **2023**, *15*, 1692. <https://doi.org/10.3390/sym15091692>

Academic Editors: Xi Xia, Yi Gao, Martiros Khurshudyan and Christos Volos

Received: 12 July 2023

Revised: 26 August 2023

Accepted: 31 August 2023

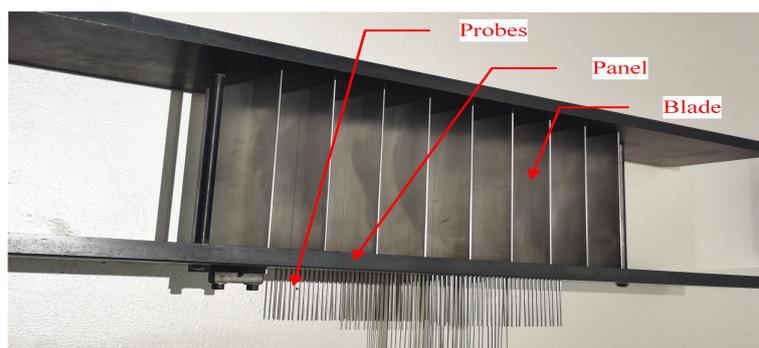
Published: 4 September 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

The blade shape in an aero-engine compressor is usually either symmetric or asymmetric, and the traditional symmetric blade shapes include the NACA series. As shown in Figure 1, the plane cascade consists of either a symmetrical shape or an asymmetrical blade shape. In order to obtain the advantages and disadvantages of symmetric or asymmetric blade shapes, plane cascades are usually used for testing. The aerodynamic coefficients of the plane cascade can reflect the pros and cons of symmetric and asymmetric blade shapes in a compressor, which in turn determines the performance of the compressor. As the core component of the aero-engine, the compressor can directly affect the aero-engine parameters, such as thrust-to-weight ratio, air flow rate, and efficiency. The geometry structure and test method of the plane cascade test piece determine the importance of the axial velocity density ratio (AVDR) parameter in determining the duality of the flow field and the validity of the data at the early stage of the test, and the value of AVDR is generally maintained at about 1 [1]. In subsonic and transonic tests, AVDR has a greater influence on the cascade loss coefficient (measurement of pressure, energy losses) and airflow turning angle [2].



**Figure 1.** Plane cascade test piece.

The aerodynamic performance of traditional plane cascades can be obtained by theoretical analysis, computational fluid dynamics (CFD), and wind tunnel tests. Theoretical analysis is based on the laws of physics and deduced through the control equations, which will encounter great difficulties in dealing with complex problems, whereas wind tunnel tests have the problems of long test periods and high costs. CFD uses computer and the laws of physics to solve equations and has achieved excellent results in flow field prediction, control optimization, and turbulence modeling [3,4], but it consumes a lot of computing resources and takes a long time when performing complex calculations. In order to speed up the CFD time, machine learning and its combination have achieved good results in grid solving, geometric modeling, flow field prediction, and pressure distribution prediction [5–9]. However, limited by the computing resources required by CFD itself, the computing time and resource consumption of problems are still a major difficulty.

As artificial intelligence has matured, data-driven machine learning methods have been widely used in aerospace and other fields, such as target detection [10]. In research on aerodynamic performance prediction, commonly used machine learning methods include support vector regression (SVR) [11,12], Gaussian process regression (GPR) [13], Kriging [14,15], and neural networks [16–22]. For classical machine learning methods, Andrés-Pérez et al. [11] and Peng et al. [12] used SVR for airfoil and rocket aerodynamic performance prediction, respectively. Experimental results show that SVR has a stronger ability to fit nonlinear data. For GPR, the kernel has a decisive influence. Different kernels have different fitting abilities for data. Hu et al. [13] proposed a GPR based on automatic kernel construction, selecting different kernels according to the set construction rules and combining kernels, avoiding the empirical error caused by manual selection. Kriging, which is essentially similar to GPR, also has the same advantages in the task. Han et al. [14] and Zhao et al. [15] proposed a Kriging based on gradient and second-order forms, respectively.

The neural network methods studied in the field of aerodynamic coefficient prediction are BPNN [17,20,23], CNN [16,18,21,24–26], PINN, and MTLNN. BPNN is generally used for pure numerical data, and CNN is used for image data. Due to the multi-dimensional output characteristics of aerodynamic coefficients, Lin et al. [19] and Zhang et al. [27] utilized MTLNN to expand different aerodynamic coefficients and data forms into different tasks for the purpose of parameter sharing between different outputs and data forms and embedded physical knowledge to form PIMTLNN.

For the methods described above, the neural network and GPR have better prediction accuracy than the other methods, but the neural network model requires a large number of datasets to obtain excellent prediction results, and the plane cascade data collection has the problems of a long test period, high input cost, and low efficiency. GPR has great advantages over neural networks for small samples. However, the traditional single-output Gaussian process regression is limited by its own functional characteristics [28] and can only model each output dimension separately in the aerodynamic coefficient prediction task. For plane cascade data with the correlation between output dimensions, single-output Gaussian process regression not only takes a long time to model, but also cannot fit the correlation between multi-dimensional outputs. Multi-output Gaussian process regression

can consider the correlation between multi-dimensional outputs, but there are two different types of input parameters in the plane cascade data, geometry and working conditions, and the traditional Euclidean distance-based sample similarity measure is difficult to improve the prediction accuracy and generalization performance of the model in the case of small samples.

For this problem, metric learning for multi-output Gaussian process regression is proposed. ML\_MOGPR combines metric learning to learn a new embedding space of sample features, which can better distinguish different samples in the task of predicting the aerodynamic coefficients of plane cascade with fewer samples. In the new metric space, the input features corresponding to each output target can have better weight ratios.

The experimental results of the aerodynamic coefficient prediction of the plane cascade show that the multi-output model is better than the single-output model in overall prediction accuracy. ML\_MOGPR further outperforms the rest of the multi-output models, backpropagation neural network (BPNN), multi-task neural network (MTLNN), and traditional MOGPR, in terms of overall prediction accuracy, which verifies the effectiveness of ML\_MOGPR in the prediction of aerodynamic performance of a plane cascade, and it can provide an important reference for the preliminary estimation of aerodynamic coefficients of the plane cascade. The main symbols and abbreviations used in this paper are shown in Table 1.

**Table 1.** Description of main variable symbols and abbreviations.

Symbol	Description	Symbol	Description
$D$	Training dataset	$\ell$	Length-scale
$\varepsilon_i$	Noises	$\nu$	Kernel scale parameter
$k(x, x')$	Covariance function	$\tau$	Distance calculation parameters
$B_q$	Positive semi-definite matrix	$a_{rq}^{(i,j)}$	Amplitude parameters
$X \in \mathbf{R}^{n \times d}$	Input features	$\varphi_q^{(i,j)}$	Displacement parameters
$Y \in \mathbf{R}^{n \times 1}$	Output targets	$\phi_q^{(i,j)}$	Delay parameters
$X_* \in \mathbf{R}^{n \times d}$	Test features	$Q$	Components
$A \in \mathbf{R}^{d \times d}$	Metrics matrix	$R_q$	Subcomponents
$\gamma$	Mounting angle/(°)	$t$	Raster distance/(mm)
$bax$	String length/(mm)	$Nob$	Number of blades
$\beta_1$	Inlet airflow angle/(°)	$Ma$	Mach number
$\alpha$	Angle attack/(°)	$\omega$	Loss coefficient
$\Omega$	AVDR	$\sigma_i^2$	Variance

## 2. Gaussian Process Regression (GPR)

### 2.1. Single-Output Gaussian Process Regression (SOGPR)

GPR is a non-parametric model that uses GP prior for regression analysis. According to the prior assumption and likelihood distribution, the posterior probability distribution of the predicted sample is obtained by the Bayesian rule.

Assuming a latent function  $f(x)$  with a Gaussian prior, its mean  $m(x)$  and covariance  $k(x, x')$  are as follows:

$$m(x) = E[f(x)] \quad (1)$$

$$k(x, x') = E[(f(x) - m(x))(f(x') - m(x')))] \quad (2)$$

The Gaussian process:

$$f(x) \sim GP(m(x), k(x, x')) \quad (3)$$

In a regression task, suppose the training dataset  $D = \{X, Y\} = \{x_i, y_i\}_{i=1}^n$ ,  $X \in \mathbf{R}^{n \times d}$  is the input matrix of  $n \times d$ , and  $Y \in \mathbf{R}^{n \times 1}$  is the output vector of  $n \times 1$ ;  $x_i \in \mathbf{R}^d$  is the

$d$ -dimensional vector, and  $y_i$  is the output scalar corresponding to  $x_i$ . There is generally noise in real datasets, i.e.,  $y_i = f(x_i) + \varepsilon_i$ , where  $\varepsilon_i$  is assumed to follow an independent and identically distributed Gaussian distribution with mean 0 and variance  $\sigma_i^2$ :  $\varepsilon_i \sim \mathcal{N}(0, \sigma_i^2)$ . A GP is defined as a collection of random variables, for convenience of computation, assuming  $m(x) = 0$ , then  $Y \sim \mathcal{N}(0, K(X, X) + \sigma_n^2 I)$ , where  $I$  is the unit matrix,  $K(X, X)$  is the covariance matrix of  $N \times N$  with  $K_{ij} = k(x_i, x_j)$ . If the test set input is  $X_* \in \mathbf{R}^{n \times d}$ , the expected prediction value is  $Y_*$ , and since any finite number in a GP has a joint Gaussian distribution, the joint prior Gaussian distribution under the independent assumption is as follows:

$$\begin{bmatrix} Y \\ Y_* \end{bmatrix} \sim \mathcal{N}\left(0, \begin{bmatrix} K(X, X) + \sigma_n^2 I & K(X, X_*) \\ K(X_*, X) & K(X_*, X_*) \end{bmatrix}\right) \quad (4)$$

The joint posterior distribution can be obtained:

$$\begin{aligned} Y_* | X_*, Y, X &\sim \mathcal{N}(\bar{Y}_*, \text{cov}(Y_*)) \\ \text{w.r.t } \bar{Y}_* &= K(X_*, X) \left[ K(X, X) + \sigma_n^2 I \right]^{-1} Y \\ \text{cov}(Y_*) &= K(X_*, X_*) - K(X_*, X) \left[ K(X, X) + \sigma_n^2 I \right]^{-1} K(X, X_*) \end{aligned} \quad (5)$$

The simplified expression  $K(X, X) = K$ ,  $K(X_*, X_*) = X_*$ , where  $\bar{Y}_*$  and  $\text{cov}(Y_*)$  are the predicted value and covariance matrix of  $Y_*$ , respectively, and the covariance represents the uncertainty in the prediction results. If  $x_* \in \mathbf{R}^{d \times 1}$  is a sample in test set  $X_*$ , then the predicted value  $y_* = \sum_{i=1}^n \alpha_i k(x_i, x_*)$ ,  $\alpha = (K + \sigma_n^2 I)^{-1} Y$ . As can be seen, GPR is mainly determined by the covariance, which is also called a kernel function. This kernel controls the covariance and similarity between any two samples.

Based on the SOGPR definition, the model itself is compatible with multiple inputs, but it is not capable of achieving multi-output target prediction, and it is not possible to jointly consider the aerodynamic performance coefficients of a multi-output plane cascade. The lack of correlation between the multiple output dimensions of the model affects the overall accuracy. Therefore, the multi-output Gaussian process regression model has advantages in this respect and can more accurately predict the aerodynamic performance coefficient of the multi-dimensional cascade.

## 2.2. Multi-Output Gaussian Process Regression (MOGPR)

The crucial aspect of Gaussian process regression is the kernel function selection and design. Earlier MOGPR considered each output individually as a Gaussian process with a Gaussian prior latent function and computed the covariance between different output dimensions by linearly combining the latent GP of each output dimension [29,30]. Assume  $y \in \mathbf{R}^{1 \times p}$  is a  $p$ -dimension output vector.

$$K = \sum_{q=1}^Q B_q \otimes k_q(x, x') = \begin{bmatrix} k^{(1,1)}(x, x') & \dots & k^{(1,p)}(x, x') \\ \vdots & \ddots & \vdots \\ k^{(p,1)}(x, x') & \dots & k^{(p,p)}(x, x') \end{bmatrix} \quad (6)$$

where  $Q$  is the number of components, and  $B_q$  is a  $p \times p$  positive semi-definite matrix of the multi-output kernel product, which represents the correlation between the outputs, also called the coregionalization matrix. The operator  $\otimes$  is a Kronecker product, and  $k^{(i,j)}$  represents the covariance of the  $i$ th and  $j$ th outputs.

The limited cross-covariance leads to a conflict with the  $\ell$  of the SOGPR kernel, and there is no explanation for the correlation of multiple outputs. The kernel Fourier transform-based spectral mixture kernel (SM) [31] generates phase shifts through a linearly weighted combination of spectral Gaussian kernels (SG), which can give explicit covariance relations and solve the conflict problem.

$$k_{SG}(\tau) = \exp\left(-\frac{\nu\tau^2}{2}\right) \cos(\mu\tau) \quad (7)$$

$$k_{SM}(\tau) = \sum_{q=1}^Q \omega_q k_{SG}(\tau; \theta_q) \quad (8)$$

where  $\tau = \|x - x'\|$ ,  $\theta = \{\mu, \nu\}$  is the kernel parameter,  $\mu$  is the peak frequency,  $\nu$  is the scale parameter, and  $\omega_q$  is the relative contribution of each SG. The combination of SM and linear model of coregionalization (LMC) forms the basic spectral mixture multi-output model (SM\_LMC), which can represent any combination of stationary kernels and better explain the relationship between different channels.

$$K_{SM\_LMC}(\tau; \theta) = \sum_{q=1}^Q B_q K_{SG}(\tau; \theta_q) \quad (9)$$

In addition, the SM-based extended multi-output kernels are cross-spectral mixture kernel (CSM) [32], multi-output spectral mixture kernel (MOSM) [33], and multi-output harmonizable spectral mixture (MOHSM) [34] proposed based on MOSM.

$$k_{CSM}(\tau) = \sum_{q=1}^Q \sum_{r=1}^{R_q} \sqrt{a_{rq}^{(i,j)}} \exp\left(-\frac{\nu_q \tau^2}{2}\right) \cos(\mu_q + \phi_{rq}^{(i,j)}) \quad (10)$$

$$k_{MOSM}(\tau) = \sum_{q=1}^Q a_q^{(i,j)} \exp\left(-\frac{\nu_q \|\tau + \phi_q^{(i,j)}\|^2}{2}\right) \times \cos\left(\left(\tau + \phi_q^{(i,j)}\right)^T \mu_q + \phi_q^{(i,j)}\right) \quad (11)$$

$$k_{MOHSM}(\tau) = \sum_{p=1}^P \sum_{q=1}^Q a_q^{(i,j)} \exp\left(-\frac{\nu_q \|\tau + \phi_q^{(i,j)}\|^2}{2}\right) \times \cos\left(\left(\tau + \phi_q^{(i,j)}\right)^T \mu_q + \phi_q^{(i,j)}\right) \times \exp\left(-\frac{l_p^2 \|\bar{\tau} - x_p\|^2}{2}\right) \quad (12)$$

where  $a_{rq}^{(i,j)}$  and  $\phi_q^{(i,j)}$  are amplitude and displacement parameters;  $R_q$  is the number of subcomponents;  $\phi_q^{(i,j)}$  are the delay parameters;  $\bar{\tau} = \|x + x'\|$ ,  $P$  is the input displacement number,  $x_p$  is the input component, and  $l$  represents the length scale parameter. The SM extension-based method makes MOGPR smoother and is able to do joint consideration of the multidimensional aerodynamic coefficients of the plane cascade, which makes the model more generalizable.

### 3. Metric Learning for MOGPR

Metric learning has been shown to have significant benefits in areas such as image classification [35–37] and regression prediction, as it is able to measure sample similarity from different perspectives in a new embedding space and shows great benefits with fewer samples. In the traditional multi-output Gaussian process regression, the sample similarity calculation form is  $\|x - x'\|$ , which gives the same weight to the input features. The input features of plane cascade data include two different types of parameters, geometry and working conditions, and the degree of influence on the aerodynamic coefficient has non-point-to-point characteristics.

As shown in Figure 2, the correlation strength between the plane cascade data features and coefficients is different. For example, the correlation between the geometrical parameter raster distance ( $t$ ) and the aerodynamic coefficient  $\Omega$  is  $-0.07$ , whereas the correlation between the working condition parameter inlet airflow angle ( $\beta_1$ ) and  $\Omega$  is  $-0.85$ . It is evident that there is a large difference between the influence of geometric and working condition parameters on the aerodynamic coefficients.

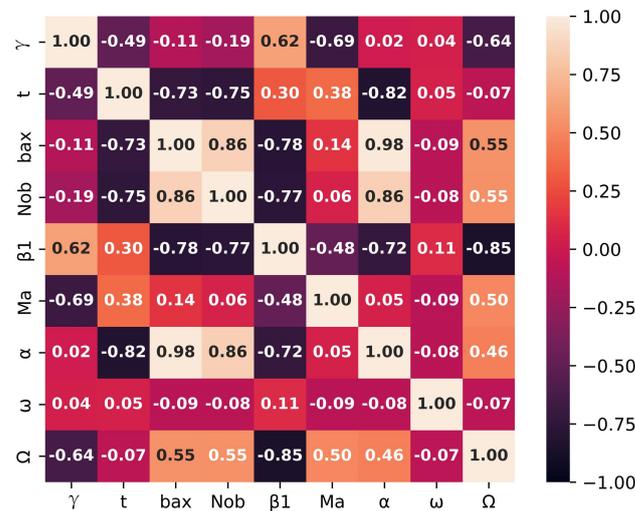


Figure 2. Plane cascade data feature correlation heat map.

In the case of small samples, the generalization ability of the model will be reduced. Inspired by [38,39], the input features are embedded into a new space, denoted as:  $x \rightarrow Ax$ , where  $x \in \mathbf{R}^{d \times 1}$ , then  $A \in \mathbf{R}^{d \times d}$ . The new distance metric formula is as follows:

$$d(x - x') = \|A(x - x')\|^2 \tag{13}$$

The model learns this new matrix by which to linearly project the original sample space. This new matrix is able to assign different weight ratios to the sample features based on the output coefficients, reducing the influence of uncorrelated features on the resulting predicted coefficients. In addition, different output coefficients have different focused features (i.e., the same input features of different output coefficients have different weights in single output modeling) in order to maintain different ratios of feature weights while further fitting the correlation between multiple output coefficients. Reference [40] has an embedding matrix  $A$  with the following form in the multi-output Gaussian kernel:

$$A^{ij} = A^i \cdot A^j \tag{14}$$

where  $(\cdot)$  is the matrix matmul product, and  $i, j$  denote the output dimensions. Equivalent to  $A^i, A^j$  is the subembedding matrix about the  $i$ th and  $j$ th dimensional output coefficients, and  $A^{(i,j)}$  is the joint matrix.

As MOSM and MOHSM are the better multi-output Gaussian kernel functions in the current study, these two kernel functions are explored in this paper, and their functional forms based on metric learning are as follows:

$$k_{ML\_MOSM}(x, x') = \sum_{q=1}^Q a_q^{(i,j)} \exp\left(-\frac{v_q^{(i,j)} \|\beta\|^2}{2}\right) \times \cos(\beta^T \mu_q + \varphi_q^{(i,j)}) \tag{15}$$

$$\begin{aligned}
k_{ML\_MOHSM}(x, x') = & \sum_{p=1}^P \sum_{q=1}^{Q_p} a_q^{(i,j)} \exp\left(-\frac{v_q^{(i,j)} \|\beta\|^2}{2}\right) \times \\
& \cos\left(\beta^T \mu_q + \varphi_q^{(i,j)}\right) \times \\
& \exp\left(-\frac{l_p^{2(i,j)} \|\bar{\tau} - x_p\|^2}{2}\right)
\end{aligned} \tag{16}$$

where  $\beta = A_q^{(i,j)}(x - x') + \varphi_q^{(i,j)}$ . ML\_MOSM and ML\_MOHSM introduce new embedding matrices and learn new feature weight ratios while inheriting the multi-output form of the original kernel function. They not only consider the differences between the output targets, but also make the parameters of the measurement matrices of different output targets shared, which make them more generalizable. In plane cascade data with smaller samples, the model can be enhanced to assign weights to parameters with large differences, and similarity measures between different samples can be implemented in a more reasonable space.

#### 4. Experiments and Analysis

The plane cascade test data used were obtained from real wind tunnel test data from a research institute. The data included plane cascades with symmetric and asymmetric blade compositions, which have varying degrees of difficulty in predicting aerodynamic coefficients. In addition, the complexity and high cost of the experimental process resulted in a complex and sparse dataset. Finally, after data cleaning, five groups of plane cascades totaling 310 samples were selected for the experiment. The input features include four geometric and three working condition parameters, and the output coefficients are cascade loss coefficient ( $\omega$ ) and AVDR ( $\Omega$ ). The main geometric differences between the five datasets are shown in Table 2.

**Table 2.** Differences of geometric parameters of five sets.

Num	Mounting Angle/(°)	Raster Distance/mm	Number of Blades	String Length/mm
1	63.2	43.35	9	70.32
2	58.29	48	8	79.18
3	40.86	50	10	64
4	76.67	48.05	6	51.08
5	76.9	35.44	11	78.11

To explore the performance of ML\_MOGPR compared with other models in the case of small plane cascade samples, 200, 250, and 300 samples were randomly selected for the experiment. In addition, to make up for the differences caused by the random division of data and ensure the stability of the model as much as possible, each model performs ten experiments in each set of data to take the average value. The training, verification, and test set are randomly divided into 8:1:1. The regression evaluation metrics are RMSE and MAE, and the optimal values are marked in bold black.

$$RMSE = \frac{1}{n} \sum_{i=1}^n \sqrt{(e(x_i; \theta) - y_i)^2} \tag{17}$$

$$MAE = \frac{1}{n} \sum_{i=1}^n |e(x_i; \theta) - y_i| \tag{18}$$

Since ML\_MOGPR is an innovation based on MOSM and MOHSM, in the field of data-driven prediction of aerodynamic coefficients, the neural networks mainly used for pure numerical data are backpropagation neural network (BPNN) and multi-task learning neural network (MTLNN), and the other single-output models are mainly SOGPR and

SVR. Therefore, the comparative models are MOSM, MOHSM, BPNN, MTLNN, SOGPR, and SVR.

Neural networks have different network structures based on the data and loss function. Specific neural networks are described below:

(1) The form of the data in this paper is similar to [19]. The MTLNN model adopts its network structure, but the physical knowledge of the plane cascade data is implicit, the embedded physical knowledge module is not used, and only the multi-task network part is used. The task layer is changed to two layers according to the two-dimensional output characteristics of the data in this paper.

(2) BPNN has different numbers of network layers and nodes according to the amount of data. In order to ensure the relative fairness of the comparison results, the BPNN network structure will be explored under three sets of data.

(3) Since the prediction of the aerodynamic coefficient of the plane cascade is a regression task, BPNN and MTLNN train the network under the two loss functions of mean square error (MSE) and mean square absolute error (MAE). Each neural network is trained with a mini\_batch size of 30 and a network output dimension of 2.

The neural network uses the pytorch, and MOGPR, ML\_MOGPR uses the MOG-PTK [41]. The experiments were performed on a computer configured with an Intel(R) Core(TM) i7-8700 CPU @ 3.20GHz, 3.19 GHz.

#### 4.1. Neural Network Structure and Loss Function Exploration

##### 4.1.1. BPNN Experiments

For the network structure exploration of BPNN, due to the small plane cascade data, in order to avoid network overfitting and underfitting, we first start with a network structure with two hidden layers and the number of nodes (8,4). Then, the number of network layers and the number of nodes in each layer are increased sequentially, and the loss functions are, respectively, chosen as MSE and MAE. Specific experiments were performed with MSE loss function and MAE loss function and sample sizes of 200, 250, and 300.

The experimental results shown in Figures 3 and 4a,b represent the experimental results of BPNN trained under the MSE and MAE loss functions, respectively. The vertical coordinates in Figures 3 and 4 indicate the RMSE prediction value, and the horizontal coordinates in Figure 3 denote the number of network layers (e.g., 2 indicates 2 hidden layers (8,4), 3 indicates 3 hidden layers (16,8,4), and incremented accordingly). The horizontal coordinates in Figure 4 indicate the number of nodes compared to the product of the initial three-layer nodes (16,8,4) in the three-layer hidden layer (e.g., 3\_1 means (16,8,4) is the initial three-layer node number, 3\_2 means multiplying the number of nodes in 1 by 2, and the number of nodes is (32,16,8)).

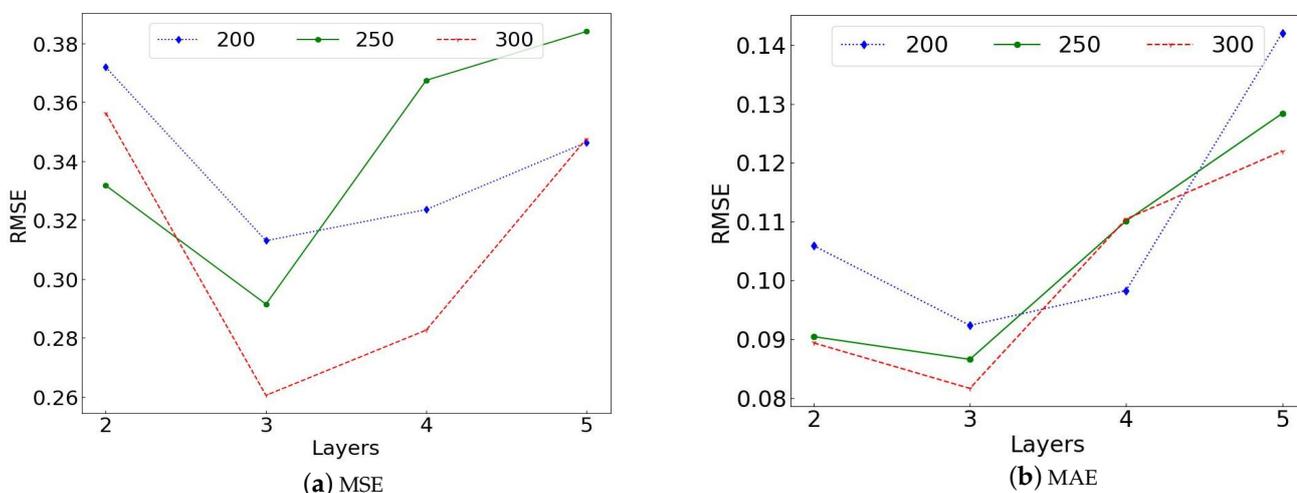


Figure 3. BPNN network layers.

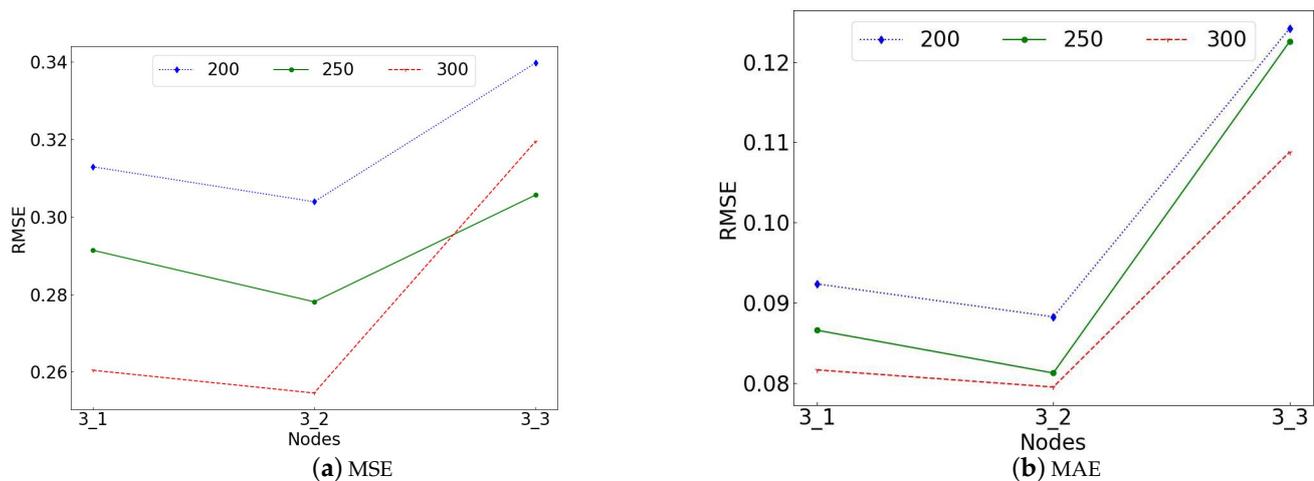


Figure 4. BPNN network nodes.

It can be seen from Figure 3 that, under the MSE and MAE, the number of hidden layers of the optimal BPNN network for the three groups of training samples is three and the network structure is (16,8,4); as shown in Figure 4, the optimal number of network nodes is (32,16,8). Refer to Table 3 for the specific values in Figures 3 and 4. It can be seen that, under the three sets of training data, the optimal values for the MSE are 0.30392, 0.27811, and 0.25457, respectively, and the optimal values for the MAE are 0.08824, 0.0814, and 0.07949. As the number of samples increases, the RMSE value is gradually decreasing regardless of the MSE or MAE. The overall prediction accuracy of MAE-guided BPNN training is far better than that of the MSE. Overall, for BPNN, the optimal network structure is (32,16,8) and the optimal loss function is MAE in the few-sample plane cascade dataset.

Table 3. BPNN network structure and loss function exploration.

Loss Function	Number of Samples	Network Structure	RMSE
MSE	200	(8,4)	0.37188
		<b>(16,8,4)</b>	<b>0.31296</b>
		(32,16,8,4)	0.32353
		(64,32,16,8,4)	0.34634
		<b>(32,16,8)</b>	<b>0.30392</b>
		(64,32,16)	0.33974
	250	(8,4)	0.33172
		<b>(16,8,4)</b>	<b>0.29141</b>
		(32,16,8,4)	0.36742
		(64,32,16,8,4)	0.38399
		<b>(32,16,8)</b>	<b>0.27811</b>
		(64,32,16)	0.30556
	300	(8,4)	0.35633
		<b>(16,8,4)</b>	<b>0.26045</b>
		(32,16,8,4)	0.28259
(64,32,16,8,4)		0.34756	
<b>(32,16,8)</b>		<b>0.25457</b>	
(64,32,16)		0.31949	

Table 3. Cont.

Loss Function	Number of Samples	Network Structure	RMSE
MAE	200	(8,4)	0.10589
		<b>(16,8,4)</b>	<b>0.09234</b>
		(32,16,8,4)	0.09827
		(64,32,16,8,4)	0.14207
		<b>(32,16,8)</b>	<b>0.08824</b>
		(64,32,16)	0.12416
	250	(8,4)	0.09043
		<b>(16,8,4)</b>	<b>0.08657</b>
		(32,16,8,4)	0.11009
		(64,32,16,8,4)	0.12838
		<b>(32,16,8)</b>	<b>0.08124</b>
		(64,32,16)	0.12254
300	(8,4)	0.08937	
	<b>(16,8,4)</b>	<b>0.08163</b>	
	(32,16,8,4)	0.11042	
	(64,32,16,8,4)	0.12196	
	<b>(32,16,8)</b>	<b>0.07949</b>	
	(64,32,16)	0.10877	

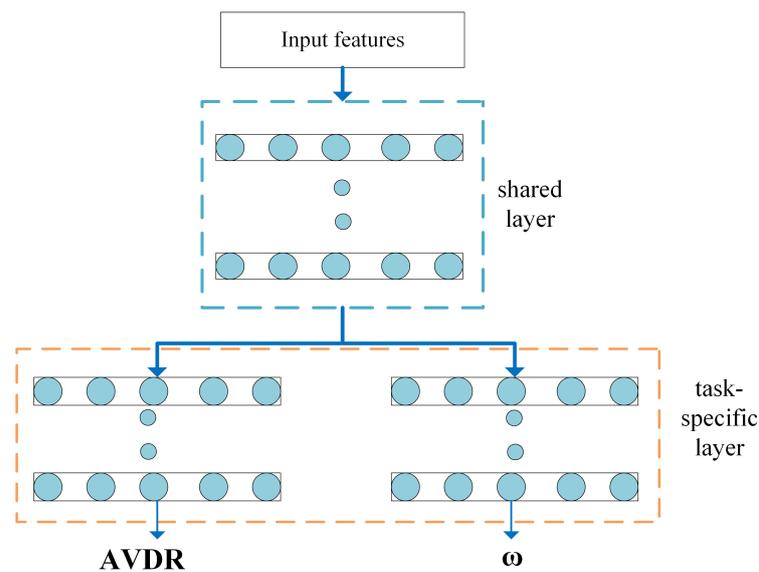
#### 4.1.2. MTLNN Experiments

The output aerodynamic coefficients of the plane cascade data have a dimension of two and do not fit into the four task layers of the MTLNN network in [19]. The shared layer of the network structure remains unchanged, and the number of task layers is changed to two. The specific MTLNN process structure is shown in Figure 5, and the detailed number of network nodes is referred to in [19]. The experimental results under the MAE and MSE loss functions are shown in Table 4. Under MSE and MAE, the RMSE value gradually decreases as the number of samples increases, and the prediction value of MAE training MTLNN is better than MSE.

Table 4. MTLNN loss function exploration.

Loss Function	Number of Samples	RMSE
MSE	200	0.63451
	250	0.51326
	300	0.40387
MAE	200	0.16859
	250	0.14789
	300	0.12659

From the training results of BPNN and MTLNN guided by MAE and MSE loss functions, the three training datasets show that for the plane cascade dataset, the MAE loss function outperforms the MSE loss function with a fewer number of samples. The reason may be that the loss coefficient of the cascade is small, usually around 0.05, while the value of the axial velocity density ratio is usually large, around 1. MSE will square the overall calculation, widening the gap between the two outputs, whereas MAE takes the absolute value directly, so under the MSE loss function, the training results of BPNN and MTLNN are inferior to the results from the MAE loss function.



**Figure 5.** MTLNN main structure flowchart.

#### 4.2. Analysis of MOGPR Parameters

MOSM and MOHSM kernels have component combinations (Equations (11) and (12)), and different numbers of component combinations will affect the final prediction results. According to the properties of MOSM and MOHSM kernel functions and their corresponding literature, it can be known that the number of components and the output dimension are generally consistent with better experimental results.

Because the number of components  $Q$  determines the count of kernels and hyper-parameters, the higher the number of parameters, the more difficult it is to optimize. From [33],  $Q$  is also called the rank of the decomposition, and the number of  $Q$  is usually less than or equal to the number of output dimensions, so the value of  $Q$  cannot be large. In this paper, the data output dimension is 2, and the number of components  $Q$  may be around 2 to make the model have an optimal value. The following will explore the specific influence of the number of components on the experimental results. The number of core components is selected as  $Q = 1, 2, 3, 4, 5$ . The  $P$  parameter (Equation (16)) in MOHSM refers to [34], which has little influence on the final result and is fixed as  $P = 1$ .

The experimental results of the specific number of MOGPR components are shown in Figure 6, where (a–d) represent the MOSM, MOHSM, ML\_MOSM, and ML\_MOHSM models, respectively, the ordinate represents the RMSE value, and the abscissa represents the number of components  $Q$ . Under the three groups of sample numbers, it can be seen from the subgraphs (a) and (c) that, for the MOSM and ML\_MOSM models, the overall prediction progress shows an upward trend with the increase of  $Q$ , and the optimal value is around  $Q = 2$ . It can be seen from the sub-figures (b) and (d) that for the MOHSM and ML\_MOHSM models, with the increase of  $Q$ , the overall RMSE value shows a downward trend. The optimal value of MOHSM is around  $Q = 4$ , and the value of ML\_MOHSM is around  $Q = 3$ . ML\_MOGPR has some volatility with  $Q$  on the MOHSM model, but the  $Q$  of ML\_MOHSM is smaller, indicating that it has fewer parameters and is easier to optimize compared to the original MOHSM model. The specific experimental results of Figure 6 are shown in Table 5.

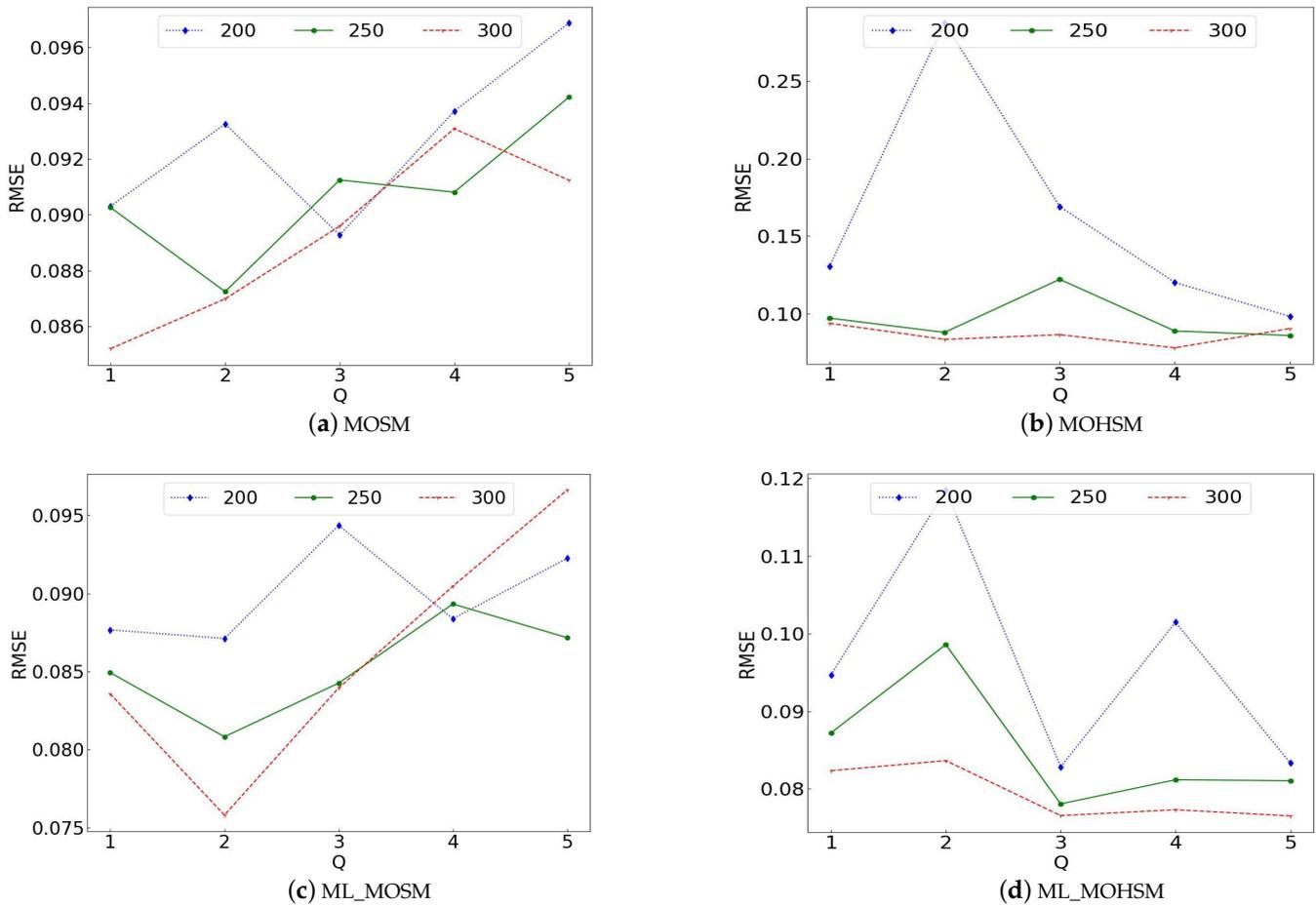


Figure 6. Analysis of the number of MOGPR components.

From Table 5, it can be seen that the experimental results of ML\_MOSM and ML\_MOHSM are better than their respective original models. The experimental results also illustrate that multi-output Gaussian process regression based on metric learning can learn a metric matrix that is more suitable for the input features of plane cascade data, avoiding the effect of large differences (As shown in Figure 2) in the features.

Table 5. MOGPR component number analysis.

Number of Samples	Model	Components				
		Q = 1	Q = 2	Q = 3	Q = 4	Q = 5
200	MOSM	0.09031	0.09326	<b>0.08927</b>	0.09372	0.09688
	MOHSM	0.13041	0.28731	0.16916	0.12024	<b>0.09843</b>
	ML_MOSM	0.08766	<b>0.08711</b>	0.09436	0.08838	0.09224
	ML_MOHSM	0.09469	0.11852	<b>0.08281</b>	0.10147	0.08331
250	MOSM	0.09026	<b>0.08724</b>	0.09125	0.09081	0.09422
	MOHSM	0.09732	0.08811	0.12228	0.08904	<b>0.08612</b>
	ML_MOSM	0.08492	<b>0.08081</b>	0.08425	0.08932	0.08716
	ML_MOHSM	0.08721	0.09859	<b>0.07804</b>	0.08115	0.08103
300	MOSM	<b>0.08519</b>	0.08698	0.08959	0.09309	0.09124
	MOHSM	0.09401	0.08361	0.08665	<b>0.07821</b>	0.09062
	ML_MOSM	0.08356	<b>0.07579</b>	0.08394	0.09046	0.09662
	ML_MOHSM	0.08232	0.08362	0.07653	0.07729	<b>0.07649</b>

### 4.3. Analysis of Results

In the above Sections 4.1 and 4.2, BPNN, MTLNN, and MOGPR have been analyzed under the RMSE evaluation metrics, respectively. For the prediction of the aerodynamic coefficient of the plane cascade, the optimal guidance function of BPNN and MTLNN networks is MAE, and the optimal network structure of BPNN is (32,16,8).

In the final comparison experiment, under three groups of sample sizes, MAE is used for the training loss function of BPNN and MTLNN, and (32,16,8) is used for the BPNN network structure. The number of MOSM, ML\_MOSM, MOHSM, and ML\_MOHSM components are, respectively, set to the component counts corresponding to the optimal RMSE values in Table 5. Tables 6 and 7 show the comparison of the best RMSE and MAE values of SOGPR (SOGPR uses a radial basis function kernel (RBF)), SVR, BPNN, MTLNN, MOSM, ML\_MOSM, MOHSM, and ML\_MOHSM, respectively.

**Table 6.** Comparison of RMSE values of different models.

Samples	SOGPR	SVR	BPNN	MTLNN	MOSM	MOHSM	ML_MOSM	ML_MOHSM
200	0.8676	5.2402	0.08824	0.16859	0.08927	0.09843	0.08711	<b>0.08281</b>
250	0.8289	4.6859	0.08124	0.14789	0.08724	0.08612	0.08081	<b>0.07804</b>
300	0.6487	4.2881	0.07949	0.12659	0.08519	0.07821	<b>0.07579</b>	0.07649

**Table 7.** Comparison of MAE values of different models.

Samples	SOGPR	SVR	BPNN	MTLNN	MOSM	MOHSM	ML_MOSM	ML_MOHSM
200	0.6241	0.8801	0.08976	0.12591	0.087561	0.06565	0.07405	<b>0.04776</b>
250	0.6101	0.7095	0.08876	0.10143	0.07246	0.05216	0.07126	<b>0.04671</b>
300	0.5736	0.6028	0.08347	0.09629	0.06635	0.05024	0.06284	<b>0.04452</b>

As can be seen from Tables 6 and 7, the multi-output model outperforms the single-output prediction of SOGPR and SVR in overall prediction, with SVR showing an overall higher RMSE value. The probable reason for this is that SVR's prediction of nonlinear data has extremely noisy points and it is poorly adapted to the plane cascade data. In addition, RMSE is the root mean square error, which goes through the sum of squares, so it exhibits a larger RMSE value. It is known that the single-output model does not fit the relationship between the output dimensions well in multi-output tasks. As can be seen in Figure 2, there is some correlation between features and outputs. The single-output model lacks some ability to fit this multi-output regression prediction of aerodynamic coefficients.

In the case of fewer samples, MTLNN performs worse and BPNN predicts better. Traditional MOSM and MOHSM have slightly worse RMSE values than BPNN with fewer samples, but better MAE values than BPNN. ML\_MOSM and ML\_MOHSM are better than BPNN in terms of RMSE value and MAE value. The shortcomings of traditional multi-output Gaussian process regression based on the Euclidean distance measure of sample similarity make it difficult to accurately measure the relationship between different samples when there are large differences in the input features of plane cascade data.

Multi-output Gaussian process regression based on metric learning should learn a new sample embedding space. In addition, according to different output dimensions with joint embedding, ML\_MOGPR is able to combine multi-dimensional embedding space based on the joint consideration of the relationship between multi-output coefficients and to assign different weights to each feature according to the relationship between the outputs, thus improving the generalization ability of the model under smaller samples. In addition, ML\_MOGPR performs better in the task of predicting the aerodynamic coefficients of symmetric and asymmetric blades of plane cascade in the case of small samples compared to other models.

## 5. Conclusions

For the shortcomings of multi-output Gaussian process regression based on the Euclidean distance measure of sample similarity in the task of predicting the aerodynamic coefficients of the small-sample plane cascade, metric learning for multi-output Gaussian process regression is proposed.

There are experimental results that show that the single-output model is worse than the multi-output model. When ML\_MOGPR is compared with its original MOSM and MOHSM models, ML\_MOGPR experimental results are better. This indicates that ML\_MOGPR should learn a new metric space in which to distinguish large difference features and assign different weight ratios and effectively improve the accuracy of MOGPR. Additionally, ML\_MOGPR outperforms BPNN and MTLNN, which shows that the proposed method can be used for a few samples of plane cascade data.

ML\_MOGPR can be applied to the preliminary estimation of plane cascade coefficients, provide a reference for the design of plane cascades, and speed up its design and test process. Further work in the future will incorporate more input features and output coefficients.

**Author Contributions:** Methodology, L.L.; Validation, L.L. and J.L.; Formal analysis, C.Y.; Investigation, L.L.; Resources, H.X. and J.L.; Writing—original draft, L.L.; Writing—review and editing, L.L. and C.Y.; Supervision, C.Y. and H.X. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by Advanced Aviation Power Innovation Workstation Project (No. HKCX2022-01-022.)

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The researcher claims no conflict of interests.

## References

1. Qingdian, Z.; Hongwei, M.; Yi, Y.; Anqi, X. Progress and prospect of aerodynamic experimental research on linear cascade. *Chin. J. Theor. Appl. Mech.* **2022**, *54*, 1755–1777.
2. Daijun, L.; Qiulin, D.; Rongchuan, Z.; Hui, W.; Jiantong, Z. Review of the cascade experimental technology. *J. Exp. Fluid Mech.* **2021**, *35*, 30–38.
3. Weiwei, Z.; Jiaqing, K.; Yilang, L. Prospect of artificial intelligence empowered fluid mechanics. *Acta Aeronaut. Astronaut. Sin.* **2021**, *42*, 524689.
4. Weiwei, Z.; Linyang, Z.; Yilang, L.; Jiaqing, K. Progresses in the application of machine learning in turbulence modeling. *Acta Aerodyn. Sin.* **2019**, *37*, 444–454.
5. Shuran, Y.; Zhen, Z.; Yiwei, W. Progress in deep convolutional neural network based flow field recognition and its applications. *Acta Aeronaut. Astronaut. Sin.* **2021**, *42*, 185–199.
6. Baiges, J.; Codina, R.; Castanar, I.; Castillo, E. A finite element reduced-order model based on adaptive mesh refinement and artificial neural networks. *Int. J. Numer. Methods Eng.* **2020**, *121*, 588–601. [[CrossRef](#)]
7. Hui, X.; Bai, J.; Wang, H.; Zhang, Y. Fast pressure distribution prediction of airfoils using deep learning. *Aerosp. Sci. Technol.* **2020**, *105*, 105949. [[CrossRef](#)]
8. Fukami, K.; Fukagata, K.; Taira, K. Super-resolution reconstruction of turbulent flows with machine learning. *J. Fluid Mech.* **2019**, *870*, 106–120. [[CrossRef](#)]
9. Brunton, S.L.; Noack, B.R.; Koumoutsakos, P. Machine learning for fluid mechanics. *Annu. Rev. Fluid Mech.* **2020**, *52*, 477–508. [[CrossRef](#)]
10. Barkhordari, M.S.; Tehranizadeh, M. Data-driven Dynamic-classifiers-based Seismic Failure Mode Detection of Deep Steel W-shape Columns. *Period. Polytech. Civ. Eng.* **2023**, *67*, 936–944. [[CrossRef](#)]
11. Andrés-Pérez, E. Data mining and machine learning techniques for aerodynamic databases: Introduction, methodology and potential benefits. *Energies* **2020**, *13*, 5807. [[CrossRef](#)]
12. Bo, P.; Rongmei, N.; Haidong, C. Surrogate Model Construction for Rocket Aerodynamic Discipline Based on Support Vector Machine. *Missiles Space Veh.* **2013**, *4*, 33–37.
13. Weijie, H.; Zenghui, H.; Xuejun, L. Missile aerodynamic performance prediction of Gaussian process through automatic kernel construction. *Acta Aeronaut. Astronaut. Sin.* **2021**, *42*, 524093.
14. Han, S.; Song, W.; Han, Z.; Wang, L. Aerodynamic inverse design method based on gradient-enhanced kriging model. *Acta Aeronaut. Astronaut. Sin.* **2017**, *38*, 138–152.
15. Xuan, Z.; Weiwei, Z.; Zichen, D. Aerodynamic modeling method incorporating pressure distribution information. *Chin. J. Theor. Appl. Mech.* **2022**, *54*, 2616–2626.

16. Du, Z.; Xu, Q.; Song, Z.; Wang, H.; Ma, Y. Prediction of Aerodynamic characteristics of compressor blades based on deep learning. *J. Aerosp. Power* **2023**, *38*, 2251–2260.
17. Zhang, G.; Cui, M. Prediction of missile's aerodynamic parameters based on neural network. *Aero Weapon*. **2020**, *27*, 28–32.
18. Zhaoyang, L.; Xueyuan, N.; Aobo, Z. Prediction of wing aerodynamic coefficient based on CNN. *J. Beijing Univ. Aeronaut. Astronaut.* **2021**, *49*, 674–680.
19. Lin, J.; Zhou, L.; Wu, P.; Yuan, W.; Zhou, Z. Research on rapid prediction technology of missile aerodynamic characteristics based on PIMTLNN. *J. Beijing Univ. Aeronaut. Astronaut.* **2021**, 1–15. [[CrossRef](#)]
20. Moin, H.; Khan, H.Z.I.; Mobeen, S.; Riaz, J. Airfoil's Aerodynamic Coefficients Prediction using Artificial Neural Network. In Proceedings of the 2022 19th International Bhurban Conference on Applied Sciences and Technology (IBCAST), Bhurban, Pakistan, 22–25 August 2022; pp. 175–182.
21. Zhang, Y.; Sung, W.J.; Mavris, D.N. Application of convolutional neural network to predict airfoil lift coefficient. In Proceedings of the 2018 AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Honolulu, HI, USA, 23–26 April 2018; p. 1903.
22. Peng, W.; Zhang, Y.; Desmarais, M. Spatial convolution neural network for efficient prediction of aerodynamic coefficients. In Proceedings of the AIAA Scitech 2021 Forum, Orlando, FL, USA, 8–12 January 2021; p. 0277.
23. Han, J.; Changping, D.; Zhixian, Y.; Guanghua, S.; Zheng, Y. Identification of aerodynamic parameters of flapping-wing micro aerial vehicle based on double BP neural network. *J. Comput. Appl.* **2019**, *39*, 299–302.
24. Wang, Q.; Yi, X. Aerodynamic parameters prediction of airfoil ice accretion based on convolutional neural network. *Flight Dyn.* **2021**, *39*, 13–18.
25. Hai, C.; Weiqi, Q.; Lei, H. Aerodynamic coefficient prediction of airfoils based on deep learning. *Acta Aerodyn. Sin.* **2018**, *36*, 294–299.
26. Barkhordari, M.S.; Armaghani, D.J.; Asteris, P.G. Structural damage identification using ensemble deep convolutional neural network models. *Comput. Model. Eng. Sci.* **2022**, *134*, 835–855. [[CrossRef](#)]
27. Jun, Z.; Guangbo, Z.; Yanqing, C.; Liwei, H.; Yu, X.; Wenyong, W. A multi-task learning method for large discrepant aerodynamic data. *Acta Aerodyn. Sin.* **2022**, *40*, 64–72.
28. Seeger, M. Gaussian processes for machine learning. *Int. J. Neural Syst.* **2004**, *14*, 69–106. [[CrossRef](#)]
29. Bonilla, E.V.; Chai, K.; Williams, C. Multi-task Gaussian process prediction. In *Advances in Neural Information Processing Systems*; Curran Associates, Inc.: New York, NY, USA, 2007; Volume 20, pp. 153–160.
30. Alvarez, M.; Lawrence, N. Sparse convolved Gaussian processes for multi-output regression. In *Advances in Neural Information Processing Systems*; Curran Associates, Inc.: New York, NY, USA, 2008; Volume 21, pp. 57–64.
31. Wilson, A.; Adams, R. Gaussian process kernels for pattern discovery and extrapolation. In Proceedings of the 30th International Conference on International Conference on Machine Learning, Atlanta, GA, USA, 16–21 June 2013; Volume 28, pp. 1067–1075.
32. Ulrich, K.R.; Carlson, D.E.; Dzirasa, K.; Carin, L. GP kernels for cross-spectrum analysis. In *Advances in Neural Information Processing Systems*; Curran Associates, Inc.: New York, NY, USA, 2015; Volume 28, pp. 1999–2007.
33. Parra, G.; Tobar, F. Spectral Mixture Kernels for Multi-Output Gaussian Processes. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 6684–6693.
34. Altamirano, M.; Tobar, F. Nonstationary multi-output Gaussian processes via harmonizable spectral mixtures. In Proceedings of the 25th International Conference on Artificial Intelligence and Statistics, PMLR, Valencia, Spain, 30 March–1 April 2022; Volume 151, pp. 3204–3218.
35. Zheng, W.; Wang, C.; Lu, J.; Zhou, J. Deep compositional metric learning. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 19–25 June 2021; pp. 9316–9325.
36. Elezi, I.; Vascon, S.; Torcinovich, A.; Pelillo, M.; Leal-Taixé, L. The group loss for deep metric learning. In Proceedings of the Computer Vision–ECCV 2020, Glasgow, UK, 23–25 August 2020; pp. 277–294.
37. Weinberger, K.Q.; Saul, L.K. Distance Metric Learning for Large Margin Nearest Neighbor Classification. *J. Mach. Learn. Res.* **2009**, *10*, 207–244.
38. Weinberger, K.Q.; Tesauro, G. Metric learning for kernel regression. In Proceedings of the Eleventh International Conference on Artificial Intelligence and Statistics, PMLR, Valencia, Spain, 25–27 August 2007; Volume 2, pp. 612–619.
39. Liu, W.; Xu, D.; Tsang, I.W.; Zhang, W. Metric learning for multi-output tasks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *41*, 408–422. [[CrossRef](#)]
40. Kan, S.; Zhang, L.; He, Z.; Cen, Y.; Chen, S.; Zhou, J. Metric learning-based kernel transformer with triplets and label constraints for feature fusion. *Pattern Recognit.* **2020**, *99*, 107086. [[CrossRef](#)]
41. de Wolff, T.; Cuevas, A.; Tobar, F. MOGPTK: The multi-output Gaussian process toolkit. *Neurocomputing* **2021**, *424*, 49–53. [[CrossRef](#)]

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.