Journal of Applied Fluid Mechanics, Vol. 18, No. 6, pp. 1364-1380, 2025.
Available online at www.jafmonline.net, ISSN 1735-3572, EISSN 1735-3645.
https://doi.org/10.47176/jafm.18.6.3171

Evaluating Diverse Meta-modeling Approaches for Predicting Performance Characteristics of a Twin Air Intake Based on Experimental Data

H. Amiri[1†], U. C. Kucuk[2], O. Kucukoglu[3], Y. F. Kuscu[3] and O.V. Ozdemır[3]

[1] *Aeronautical Engineering Dept. Sivas University of Science and Technology, Sivas, Turkey*
[2] *Tubitak Sage, Ankara, Turkey*
[3] *Turkish Aerospace Industries, Ankara, Turkey*

†*Corresponding Author Email: human.amiri@sivas.edu.tr*

ABSTRACT

Air intakes are critical components in maximizing the efficiency of jet-powered engines. Their diverse designs, ranging from conventional shapes to innovative configurations, coupled with the intricate interplay of fluid dynamics, boundary layer effects, and structural considerations, render the determination of their performance characteristics a time-consuming task. However, a meticulous and confident evaluation of these characteristics is the key to achieving optimal air intake design and, consequently, significant enhancement of overall engine performance. This article assesses various meta-modeling approaches for predicting the performance characteristics of a twin air intake system. A comprehensive exploration of meta-modeling methods, particularly those specifically tailored for data derived from experiments, is presented. A database of 4000 experimentally obtained runs is utilized to construct train and test data for diverse models, including polynomials, decision trees, random forest regression, multivariate adaptive regression splines, and neural networks. The performance of each model is rigorously evaluated based on goodness of fit, precision, accuracy, monotonicity, and interpretability. This study provides a cost-effective and time-efficient alternative for predicting crucial flow parameters associated with the air intake of jet engines. The results reveal that the Random Forest Regression (RFR) model outperforms all other models across all evaluated metrics, demonstrating its superior effectiveness in predicting the performance characteristics of the twin air intake system.

Article History

Received October 8, 2024
Revised December 12, 2024
Accepted January 3, 2025
Available online March 30, 2025

Keywords:

Meta-modeling
Surrogate modeling
Twin air intake
Pressure recovery
Pressure distortion

1. INTRODUCTION

The emergence of meta-models (or surrogate models) in the late 1980s (Box & Draper, 1987) revolutionized the modeling of complex systems. Since then, researchers have dedicated significant efforts to develop new methods and models to not only reduce the cost of experiments and numerical simulations but also provide means to filter raw data potentially contaminated with noise (Forrester et al., 2008). This study focuses on analyzing meta-models frequently applied in practical engineering applications, prioritizing those commonly used in both experimental and numerical (CFD) simulations (Kianifar & Campean, 2020). These models include polynomials, response surface methodology (RSM), regression splines, kriging, moving least squares (MLS), support vector regression (SVR), multivariate adaptive regression splines (MARS), radial basis function (RBF), and neural networks (NN). The performance of each model is evaluated based on accuracy, efficiency, and robustness. This study utilizes a set of experimental data for meta-modeling. Utilizing experimental data for meta-modeling offers several advantages. These models, constructed from real experimental data, are valuable as they represent real-world phenomena with fewer assumptions compared to numerical simulations (Kleijnen, 2017). However, both experimental and numerical data have unique strengths and limitations. While experimental data may provide insights into actual physical systems, it can be affected by measurement errors, noise, and instrumentation constraints. Numerical simulations, on the other hand, offer high control and repeatability but may rely on assumptions and approximations that could affect their fidelity. Hence, the choice of data source for training AI

NOMENCLATURE

Acronyms

AID	Automatic Interaction Detection
AIP	Aerodynamic Interface Plane
AoA	Angle of Attack (°)
AoSS	Angle of Side Slip (°)
BWB	Blended-wing-body
CART	Classification and Regression Trees
CCC	Concordance Correlation Coefficient
CFD	Computational Fluid Dynamics
DoE	Design of Experiment
DSM	Double-stage Metamodel
DTR	Decision Tree Regressor
GBR	Gradient Boosting Regression
GCV	Generalized Cross Validation
GUIDE	Generalized Unbiased Interaction Detection and Estimation
KNN	K-Nearest Neighbors
KRCC	Kendall's Rank Correlation Coefficient
LR	Linear Regression
MAE	Mean Absolute Error
MARS	Multivariate Adaptive Regression Splines
MFR	Mass Flow Rate
MFRC	Corrected Mass Flow Rate
MLR	Multiple Linear Regression
MLS	Moving Least Squares
MSE	Mean Squared Error
NN	Neural Networks
OLS	Ordinary Least Squares
PR	Total Pressure Recovery
RBF	Radial Basis Function
ReLU	Rectified Linear Unit
RFR	Random Forest Regression
RMSE	Root Mean Squared Error
RSM	Response Surface methodology
SCC	Spearman correlation coefficient
SVR	Support Vector Regression
UAV	Unmanned Aerial Vehicles

Roman Symbols

$P_{t_{AIP}}$	area weighted total pressure at AIP
P_{t_∞}	available total pressure at free stream
P_{ref}	pressure reference (101325 pa)
P_θ	averaged total pressure of a circular sector of θ degrees which has the highest total pressure deficiency (pa)
T_{ref}	temperature reference (288.15k)
X_i, x_i	independent or predictor variables ($i = 1,2,\dots,p$)
a_i	coefficients ($i = 1,2,\dots,p$)
$f_i(X)$	base function ($i = 1,2,\dots,p$)
q_{AIP}	dynamic pressure at aip (engine face) (pa)
$\overline{y}$	mean of actual values
N	number of observations, number of nodes in the neural network
R^2	r-squared
BF(x)	basis function
DC60	total pressure distortion coefficient over a 60° sector of aip
DCθ	total pressure distortion coefficient over a θ degree sector of aip
L	number of layers in neural network
M	Mach number
P	pressure (pa)
T	temperature (k)
Var( )	variance
$Y, \hat{y}$	predicted or expected value of the dependent variable
cov(X, Y)	covariance between the predicted values (x) and the true values (y)
enp	number of basis functions
j	number of independent variables
n	number of input features
p	order of polynomial
ref	reference value
t	total (stagnation) value
y	actual (true) value

Greek Symbols

β_0, β_n	constant and coefficient of the ith basis function
Θ	temperature ratio (t_t/t_{ref})
δ	pressure ratio (p_t/p_{ref})
ϵ	residual (error of the regression)

models depends on the specific context and objectives of the study.

Experimental data-based models can be used to model real-world systems, enhancing their applicability to practical engineering problems (Kianifar & Campean, 2020) and fostering a deeper understanding of the system (Kleijnen, 2009). They can also prove more efficient and cost-effective when simulations demand high computational resources (Simpson et al., 2001). The application of metamodeling for optimizing air intake aerodynamics in jet engines has been explored in several studies, including the use of kriging models with efficient global optimization algorithms and multi-objective surrogate models (Drężek et al., 2022).

Typically, meta-modeling involves selecting an experimental design for data generation, choosing a suitable model to represent the data, and subsequently fitting the model to the data (Simpson et al., 2001). Several approaches have been proposed for meta-modeling using experimental data. Kriging models are often employed when data are obtained from a larger experimental domain (Kleijnen, 2009). Adaptive design of experiments (DoE) methodologies have been developed for global Kriging meta-modeling applications, allowing for the adaptive selection of new experimental points based on the current metamodel (Kyprioti et al., 2020). A recent algorithm for adaptive computer experiments facilitates the construction of a metamodel using large datasets (Erickson, 2019). In aerodynamic design optimization, a double-stage metamodel (DSM) has been proposed that integrates the advantages of both interpolation and regression meta-models and utilizes experimentally obtained data as input (Friedman & Pressman, 1988). In another study it has been demonstrated that a single metamodel can be deemed

reliable when evaluating an initial set of experiments from the simulator (Kleijnen, 2009).

This study aims to identify a meta-model (or multiple models) that surpasses others in predicting the performance characteristics of a specific air intake. The air intake is a twin-type model with a circular cross-section and has been subjected to prior experimental studies in a wind tunnel facility. Independent variables controlled during the experimental campaign include free stream Mach number (M), the air intake's attitude in terms of angle of attack (AoA), angle of side slip (AoSS), and mass flow rate passing the intake (MFR).

The variables M, AoA, AoSS, and MFR were chosen for this study as they capture the primary aerodynamic factors influencing total pressure recovery (PR) and circumferential distortion (DC60). Other factors, such as material properties, fabrication processes, and size, were not included as they were deemed secondary under the controlled conditions of the experimental setup. The intake model used in this study was fabricated with a smooth surface and tested at scaled conditions to maintain dynamic similarity, minimizing the influence of these factors on the flow parameters.

Pressure Recovery (PR) and the 60° Distortion Coefficient (DC60) were chosen as the output variables for this study due to their critical importance in assessing the aerodynamic performance of air intake systems. PR quantifies the efficiency of total pressure recovery, directly impacting engine thrust and efficiency. DC60 measures the circumferential non-uniformity of total pressure, which affects compressor stability and engine-intake compatibility. Together, these parameters provide a holistic assessment of the intake's ability to deliver uniform and high-energy airflow to the engine, aligning with the primary objectives of intake design in aerospace applications.

While this can be categorized as a low-dimensional problem with only four independent variables, the interaction between these variables introduces non-linearity, increasing the problem's complexity. Due to limited available experimental results, the meta-modeling conditions are constrained by a sample size with minimal information regarding data uncertainty (noise). Identifying the most suitable model for this specific flow problem has the potential to significantly reduce the time and cost required for future experimental campaigns. Additionally, prioritizing parametric models expressible as explicit functions facilitates their implementation in future applications.

This paper is organized as follows: The introductory section provides a comprehensive overview of the twin air intake under investigation, outlining the key aerodynamic performance parameters crucial for subsequent analysis. A description of commonly employed meta-modeling methods, specifically tailored for experimentally derived data, is given alongside common and less common statistical metrics used for model evaluation. The methodology section offers insights into the meticulous process of preparing the experimental database, emphasizing data normalization, scripting procedures, and systematic result extraction. The subsequent section critically evaluates and discusses the performance of various meta-models utilized in the study, shedding light on their effectiveness in capturing the intricacies of the twin air intake system. Finally, the paper concludes with the findings of this research.

1.1. Twin Air Intake

Designing an air intake system is a complex and multifaceted job, demanding consideration of diverse parameters categorized as either geometric or relating to the fluid dynamics of the flow. Beyond geometric factors, achieving optimal performance and efficiency necessitates meticulous analysis and evaluation of the internal flow. To assess the effectiveness of a specific air intake design, conducting experiments with scaled prototypes within controlled environments, such as wind tunnel facilities, is essential. These experiments offer valuable insights into the behavior of the air intake and facilitate identification of key performance indicators. Comprehensive analysis of such intricate engineering systems necessitates either an extensive and time-consuming experimental campaign or access to a high-performance computing facility with significant computational resources.

This study utilizes available experimental data collected for a specific twin air intake to analyze the total pressure recovery and its distortion at the aerodynamic interface plane (AIP). These two parameters are crucial in evaluating the performance of the air intake system, as they directly influence its efficiency and functionality (Seddon et al., 1999). Figure 1 schematically depicts the Y-shaped air intake duct commonly employed in single-engine high-speed aircraft (Patel et al., 2005; El-Sayed, 2016). Typically, the intakes are located on the sides of the fuselage, with the two sections of the duct converging within the fuselage to form a single duct. The performance of Y-shaped intakes is significantly influenced by the aircraft fuselage, prompting numerous studies to focus on examining the intake flow characteristics during various flight phases and propose methods for enhancing their performance.

In steady-state flight, characterized by zero angle of attack and sideslip angle and thus symmetric inflow conditions, each intake in a conventional Y-shaped diffuser system supplies an equal share of the engine's required air mass flow. However, recent research by Askari and Soltani (2020) identified the presence of asymmetric flow within diverterless supersonic Y-shaped

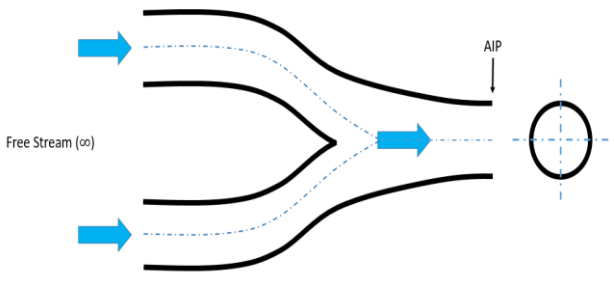


Fig. 1 Schematics of a Twin (Y-shaped) air intake

diffusers. Furthermore, inherent unsteadiness and nonzero sideslip angles (asymmetric conditions) inevitably cause unequal mass flow distribution between the left and right diffuser limbs. Despite the seemingly smooth airflow within each individual duct, significant distortion is highly likely to occur upon mixing at the aircraft integration point (AIP) (El-Sayed, 2016). Therefore, the assessment of total pressure recovery, alongside the evaluation of its circumferential distortion, assumes critical importance when examining the compatibility between the engine and the intake system. This compelling factor necessitates prioritizing these two aspects as primary objectives within the present modeling study. For brevity, a concise explanation of each factor follows.

1.2. Total Pressure Recovery

Maximizing the total pressure of the free stream air is crucial for the performance of any air-breathing engine. This responsibility falls upon the air intake, which strives to recover the highest possible pressure. The degree of total pressure recovered at the aerodynamic interface plane (AIP) is quantified by the total pressure recovery, defined as the ratio of the surface-averaged total pressure at the AIP to the free stream total pressure. This relationship is mathematically expressed by Equation (1):

$$PR = \frac{P_{t_{AIP}}}{P_{t_\infty}} \tag{1}$$

where PR denotes intake total pressure recovery, $P_{t_{AIP}}$ is the area weighted total pressure at AIP, and P_{t_∞} is the available total pressure at free stream (Seddon et al., 1999). PR can range between 0 and 1 with the maximum value of 1 which shows 100% of total pressure recovered at AIP. Total pressure values are normally obtained by a circular rake which consists of an array of tubes equally distributed over AIP as illustrated in Fig. 2.

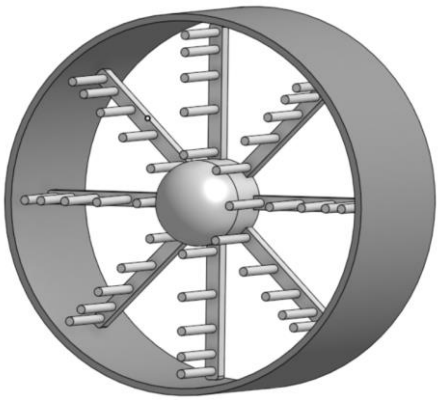


Fig. 2 Schematics of a Pressure rake for total pressure acquisition

1.3. Total Pressure Circumferential Distortion Coefficient

It is possible to evaluate the distortion of different flow variables (temperature, pressure, and velocity) over the AIP. The distortion in velocity, commonly referred to as swirl, and deviations in temperature are not the focus of this study and are therefore excluded from the analysis. The total pressure distortion can be expressed either circumferentially or radially. In this study, only the circumferential distortion is introduced which is obtained by (Seddon et al., 1999):

$$DC\theta = \frac{P_{t_{AIP}} - P_\theta}{q_{AIP}} \tag{2}$$

This is one of the many expressions used in the literature to define the deviation of total pressure. In this equation, q_{AIP} is the dynamic pressure at the engine face (AIP) and $P_{t_{AIP}}$ is the average of all total pressure readings over AIP. P_θ is the averaged total pressure of a circular sector of θ degrees which has the highest total pressure deficiency. Total-pressure contours and θ sector for definition of distortion coefficient is schematically shown in Fig. 3. If the case of θ=60°, the calculated distortion coefficient is known as DC60. The 60° sector distortion coefficient (DC60) was chosen for this study due to its widespread use in the aerospace industry and its relevance to assessing compatibility with modern jet engines. The 60° sector captures critical distortion effects over a representative portion of the compressor face, making it an industry-standard metric. While other sector angles (e.g., 30° or 45°) could provide additional insights, they were not included in this study to maintain consistency with established practices and facilitate comparison with existing literature.

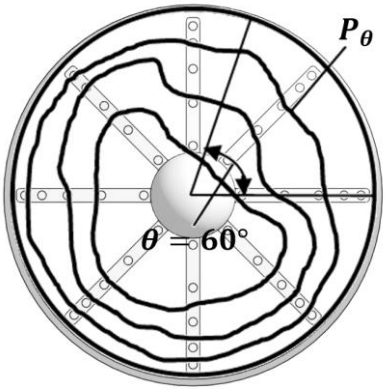


Fig. 3 60° sector for calculating DC60 distortion coefficient (Reproduced from Seddon et al. 1999)

Numerous flow parameters are known to influence the performance metrics of air intake. This study considers the free stream Mach number (M), angle of attack (AoA), angle of side-slip (AoSS), and the mass flow rate passing the aerodynamic interface plane (MFRC) as the input factors impacting the output variables: Pressure Ratio (PR) and Distortion Coefficient 60 (DC60). Therefore, the objective is to develop a meta-model that optimally fits this multivariate problem, as depicted in Fig. 4.

It is noteworthy that the corrected mass flow rate (MFRC) is obtained by:

$$MFRC = MFR.\frac{\sqrt{\Theta}}{\delta}, \qquad \Theta = \frac{T_t}{T_{ref}}, \delta = \frac{P_t}{P_{ref}} \tag{3}$$

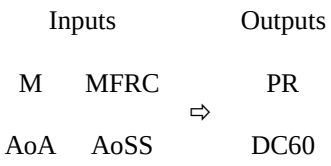


Fig. 4 Inputs and Outputs of the problem

The reference temperature (T_{ref}) and pressure (P_{ref}) values are 288.15 K and 101325 Pa, respectively. Additionally, T_t and P_t represent the area-averaged total temperature and pressure at the aerodynamic interface plane (AIP), respectively. These values, along with the pressure readings from the total rake and static ports distributed across the inner walls of the intake at the AIP plane, are used to calculate the mass flow rate (MFR) passing the AIP. Typically, a blockage (mass flow plug) located at the back end of the duct regulates the mass flow rate.

The data employed in this study originates from previously conducted experiments, where a series of discrete points were collected to characterize the air intake's behavior under various operating conditions. The database comprises approximately 4000 data points, each representing a unique combination of the below-mentioned independent input variables. The comprehensive dataset is believed to be acquired across a broad spectrum of angle of attack, angle of sideslip, free stream Mach number, and adjusted mass flow rate at the air intake, enhancing confidence in the generalizability of the results. However, design of Experiment (DoE) methodologies are often employed to generate a cost-effective database with the minimal number of experiments required. The present database originates from previous research activities conducted without utilizing DoE techniques. Consequently, the current sample set may potentially contain noise or fail to comprehensively represent the true behavior of the air intake. This necessitates the development of a suitable model that can accurately predict the air intake's performance characteristics at unknown points, i.e., points not covered by the measured data, effectively filling the gaps within the existing experimental data. Such a model is known as a "surrogate model" or "meta-model" in the engineering community (Kianifar & Campean, 2020). This study aims to fulfill this requirement by constructing a robust and reliable model capable of accurately estimating the air intake's behavior beyond the limitations of the available experimental data. By leveraging advanced mathematical and statistical techniques, this model will offer valuable insights into the performance characteristics of the air intake system. Furthermore, it will equip engineers and designers with a powerful tool for optimizing the design process, identifying areas for improvement, and ultimately enhancing the overall performance of air intake systems.

1.4. Meta-Modeling

Meta-modeling refers to the construction of surrogate models or statistical models that approximate the behavior of a complex system or process based on a limited set of input-output data. In the context of data analysis, meta modeling and surrogate modeling are interchangeably employed (Kianifar & Campean, 2020), with meta modeling commonly associated with simulating a model. Nevertheless, both methodologies are considered valid for analyzing either experimentally driven or simulated data. The surrogate model is then used to predict the system's behavior or response for new input conditions, without the need for computationally expensive or time-consuming simulations or experiments. The challenge here is to find the most appropriate model for a particular physical system; in other words, there is no such universal model that fits to all physical phenomena. In the essence of aerodynamics of air intakes, the physical parameters are pressure, temperature, velocity, and their distribution. The aim here is to focus on the total pressure and its distortion level. Hence, the model which will hopefully best fits for this specific problem should be certainly reevaluated for other parameters as well as physical problems.

Metamodeling techniques are typically categorized into two types: parametric and non-parametric ones. Parametric models explicitly rely on the underlying structure of the model. On the other hand, non-parametric methods (e.g. Radial Basis Function (RBF), Neural networks, Decision Trees, etc.) utilize experimental measurements to establish the relationship among the parameters without a need of explicit assumptions about the model. In this study we compare between parametric and non-parametric models. We start from the simplest ones (linear Regressor) and dive into more complicated ensemble models like Random forest Regressor and Gradient Boosting Regression (GBR). The comparison then will continue with Decision trees and MARS models and then some Neural nets with different hidden layers.

In this study a total of nine distinct methods has been used to model the experimentally driven data set:

Linear regression; In order to implement a linear regression model on a database containing multiple input and output variables, the multiple linear regression (MLR) technique can be employed. Multiple linear regression is an advanced form of simple linear regression that incorporates multiple predictor variables to forecast the value of a response variable (Schneider et al., 2010).

Polynomials (Morris & Mitchell, 1995), with simple structures, less computational effort requirement, and better smoothing capability. On the other hand, it has been shown that they are very prone to give erroneous results in highly nonlinear and large scale problems (Kianifar & Campean, 2020).

Decision tree (Jena & Dehuri, 2020) which can be used for both classification and regression tasks. The regression decision trees differ from classification trees by incorporating values or piecewise models at their leaves instead of class labels (Kim et al., 2007). Several commonly used regression tree algorithms exist, including Automatic Interaction Detection (AID), Classification and Regression Trees (CART), Multivariate Adaptive Regression Splines (MARS), Generalized Unbiased Interaction Detection and Estimation (GUIDE), M5, and M5' (Jena & Dehuri, 2020). The decision tree model

utilized in this study is based on the Classification and Regression Tree (CART) algorithm. This approach, implemented through the Scikit-learn library in Python, constructs binary trees by splitting the data recursively based on the feature that minimizes the mean squared error (MSE) at each node.

Gradient Boosting Regression (GBR) which is an additive modeling approach where each subsequent model is built to correct the mistakes made by the previous models. The learning process is guided by the gradient descent optimization algorithm, which minimizes the loss function of the overall ensemble. It has been used for a wide range of applications such as monitoring and prediction systems (Aziz et al., 2020) and forecasting power in wind energy (Singh et al., 2021).

Random Forest Regression (RFR) which is an ensemble learning method that combines multiple decision trees to perform regression tasks. It utilizes the principles of bagging and random feature selection to build a robust and accurate regression model.

Multivariate Adaptive Regression Splines (MARS) - It is a regression technique which is completely driven by data without any underlying assumptions about the relationship between dependent and independent variables (Friedman, 1991). This algorithm has been used extensively in many engineering problems such as crashworthiness simulation (Yang et al., 2000) and optimization (Wang et al., 2011), waste water treatment (Chen et al., 2006), process simulation (Li et al., 2010) as well as building simulation (Van Gelder et al., 2014).

Radial basis functions (RBF) – RBF is a popular method used in meta-modeling for aerodynamics and aerospace engineering. It is a type of interpolation method that uses a radial basis function to approximate the relationship between input and output variables when they are nonlinear and complex. In meta-modeling, RBF is used to build a surrogate model that can predict the behavior of a complex system based on a limited set of input parameters. An example of RBF meta-modeling in aerospace is the prediction of aerodynamic coefficients for a blended-wing-body (BWB) configuration. In a recent study, researchers used RBF to build a surrogate model for the prediction of lift and drag coefficients for a BWB configuration (Yang et al., 2018). RBF is also used in combination with other methods, such as neural networks, for meta-modeling in aerospace and aerodynamics. For example, a recent study used a combination of RBF and neural networks to build a surrogate model for flight load calculation (Yan et al., 2023).

K-Nearest Neighbors (KNN) – The term "neighbors" refers to data points that are close to each other in the feature space. K-Nearest Neighbors (KNN) regression is a non-parametric method that predicts the output for a given input by averaging the outputs of its k nearest neighbors in the feature space. The choice of k significantly impacts the model's performance, with smaller values of k capturing local variability and larger values smoothing the predictions. This study evaluates the performance of KNN regression for predicting aerodynamic parameters by optimizing k through cross-validation. This algorithm classifies a new data point by identifying the majority class among its k nearest neighbors (Altman, 1992). The choice of k, the number of neighbors considered, is a crucial parameter that influences the model's performance. It uses the entire dataset for regression, making it particularly useful for applications with large amounts of data and where the decision boundary is highly irregular (Le Clainche et al., 2023). It has been used in number of applications like instance-based algorithm for making predictions (Li et al., 2022), aircraft performance improvement (Le Clainche et al., 2023), recognition of large-scale supersonic inlet flow patterns (Wu et al., 2022), and rotary-wing UAVs (Wang et al., 2019).

Neural networks (NN) – Neural networks have emerged as powerful tools in meta-modeling within engineering applications, revolutionizing the way complex systems are analyzed and optimized. These networks can efficiently capture and replicate the underlying relationships between various design parameters and performance metrics, enabling engineers to rapidly explore design spaces and identify optimal configurations. Neural networks excel in capturing the nonlinearities and intricate dependencies present in aerodynamic systems, providing a more accurate representation than traditional analytical models. This capability significantly accelerates the design iteration process and contributes to the development of more fuel-efficient and aerodynamically superior aircraft.

Neural networks are particularly well-suited to the task of meta-modeling because they can learn complex relationships between inputs and outputs, and can generalize well to new data. One example of neural network meta-modeling in aerospace is the prediction of noise radiated by an array of propellers. In a recent study, researchers used a deep neural network approach to build a surrogate model for this prediction (Poggi et al., 2022). Another example is the modeling of aerodynamic data using a convolutional neural network. This approach has been used to predict airfoil lift and drag coefficients for various shapes defined by B-spline curve variables and flight status variables (Zan et al., 2022). In a recent study on flight loads, multiple algorithms have been analyzed and it was found that neural network residual Kriging was the most accurate method for predicting flight loads (Yan et al., 2023). Other examples can be given for a study that have used neural networks for aerodynamic optimization efficiency improvement (Gabriel Pereira Gouveia da Silva, 2019), and for surrogate modeling in aerodynamic design applications (Sun & Wang, 2019).

Evaluating each of the above mentioned models necessitates the use of specific statistical metrics, some well-established in the literature and others less commonly employed. These metrics are briefly described as follows.

R-squared (R2) measures the proportion of the variance in the target variable that is predictable from the independent variables. A higher R2 indicates a better fit.

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \overline{y})^2} \tag{4}$$

Mean squared error (MSE) calculates the average squared difference between predicted and actual values.

Lower MSE indicates better performance.

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \quad (5)$$

MSE is actually the variance of the error where a high value indicates that the model is sensitive to changes in the training data and may not generalize well to new, unseen data. Root mean squared error (RMSE) is the square root of MSE. It provides a measure in the original units of the target variable, making it easier to interpret.

$$RMSE = \sqrt{MSE} \quad (6)$$

Mean Absolute Error (MAE) calculates the average absolute difference between predicted and actual values. Like RMSE, it is in the original units.

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i| \quad (7)$$

MAE and Bias error are changing similarly. In other words, a small bias or MAE indicates that, on average, the model's predictions are close to the true values. Balancing bias (MAE) and variance (MSE) is crucial for model performance.

Prediction interval provides a range within which the actual response is likely to fall with a certain level of confidence. This is useful for quantification of uncertainty in predictions in a regression problem.

Concordance Correlation Coefficient (CCC) is a measure of agreement between observed and predicted values, considering both precision and accuracy. The CCC is used to evaluate how well the predictions of a model align with the observed outcomes. The CCC is an extension of the Pearson correlation coefficient (PCC) (Pearson, 1895), and it takes into account both precision and accuracy. It measures not only how well the points fall on a line (correlation) but also how well the line fits the 45-degree line through the origin (accuracy). The formula for the Concordance Correlation Coefficient is given by:

$$CCC = \frac{2 \times cov(\hat{y}, Y)}{Var(\hat{y}) + Var(Y) + \left(\overline{\hat{y}} - \overline{Y}\right)^2} \quad (8)$$

where cov (X, Y) is the covariance between the predicted values (X) and the true values (Y), $Var(\hat{y})$, $Var(Y)$, $\overline{\hat{y}}$, and $\overline{Y}$ are the variances and averages of X and Y, respectively. The CCC ranges from −1 to 1 where 1 indicates perfect concordance (perfect agreement between predicted and observed values), 0 indicates no concordance (random agreement), and −1 indicates perfect discordance (systematic disagreement). Therefore, in the context of CCC, a bigger number is better.

Kendall's Tau (Kendall rank correlation) and Spearman's Rank Correlation are metrics which assess the monotonic relationship between predicted and actual values, which can be useful when the assumption of linearity is not met. The value for both metrics ranges from -1 to 1, where 1 indicates perfect agreement, -1 indicates perfect disagreement, and 0 indicates no correlation. Kendall's Rank Correlation Coefficient (KRCC) and Spearman correlation coefficient (SCC) are both more suitable for cases where the relationship between variables may be monotonic but not necessarily linear.

2. METHOD

The experimental data comprising 4,000 data points were obtained through a comprehensive series of wind tunnel tests conducted previously by the authors. The experiments were performed in a Tri-sonic wind tunnel. Measurements were acquired using calibrated total pressure rakes and static pressure ports, strategically placed at the aerodynamic interface plane (AIP). The free stream conditions, including Mach number (M), angle of attack (AoA), angle of sideslip (AoSS), and mass flow rate (MFR), were systematically varied to cover a broad operating envelope. Data processing techniques, including filtering and normalization, were employed to minimize noise and ensure data reliability. Details of the experimental setup, instrumentation, and procedures are excluded since they fall outside the scope of this study. The original data set comprises 4,000 observations (excluding unreliable points), presented in a general format in Table 1. Within the current context, this data represents the input-output relationship of the system under investigation, with the first four variables serving as inputs and the last two columns representing the responses or outputs.

Normalization of a dataset is a frequently required step for various machine learning models. Poor performance may occur if the individual features deviate significantly from the characteristics of standard normally distributed data. The raw input variables are then normalized by scaling the input matrix with the help of Scikit-learn preprocessing library in python. The normalization includes removing the mean and scaling to unit variance (i.e. standard scaler). The same set of training data (80%) and test data (20%) are used for all the models under investigation. Normalized range of input parameters are given in Table 2.

Table 1 The original dataset structure

	Input Variables (Factors)				Output Variables (Responses)	
Data point	Free Stream Mach Number	Angle of Attack	Sideslip angle	Corrected Mass Flow Rate passing through AIP	Total Pressure Recovery at AIP	Circumferential Distortion in Total Pressure at AIP
	M	AoA	AoSS	MFRC	PR	DC60
1						
2						
…						

Table 2 Normalized range of input parameters

Factor	Number of levels	Normalized range
AoA	15	[-1.61 +2.30]
AoSS	17	[-1.06 +2.31]
M	18	[-1.85 +1.98]
MFRC	-	[-1.81 +1.61]

The preprocessed experimental data is used to train a surrogate model, which is a mathematical model. Statistical models like Gaussian process regression and response surface methodology will remain out of the scope of this study. The surrogate model learns the patterns and relationships in the data and creates a mathematical representation of the system's behavior. The trained surrogate model is validated using a separate set of experimental data. This step ensures that the surrogate model accurately captures the system's behavior within the range of the training data. Cross-validation techniques are not used since the performance of all models has to be determined with a same set of data. Models and metrics are implemented using scikit-learn and scipy packages available in Python. For neural networks, TensorFlow is used and where necessary, user built functions are defined. The performance parameters are generally expressed in plots which are generated using matplotlib and Microsoft® Excel. Once the meta model is validated, and the performance of the models on the parameters is evaluated, the target surrogate model would be selected and can be used to predict the system's response for new input conditions. This could be a point that was not part of the original experimental data or an erroneous or suspicious measurement point.

In the initial stage of investigation, it is necessary to choose the most effective model in each of the polynomial and neural network categories. Therefore, first, a detailed evaluation of different polynomial models and neural networks was done.

In this study we use 2, 3, 4, 5, 6 and 7th order polynomials. Since there are 4 individual input variables, the complete set of terms forming a polynomial fit will be obtained as follows. The Polynomial fit package being used is PolynomialFeatures in Scikit-learn which by default sets the polynomial with degree 2 as:

- Four individual input terms (i.e. x_1, x_2, x_3, x_4) also known as linear terms
- Four squared terms (i.e. $x_1^2, x_2^2, x_3^2, x_4^2$) also known as squared terms
- six iteration terms (i.e. $x_1x_2, x_1x_3, x_1x_4, x_2x_3, x_2x_4, x_3x_4$) also known as interaction terms.

these add up to a total of 14 terms, each associated with a coefficient in the fitted polynomial equation. This is also known as full quadratic equation. A complete third order polynomial fit consists of 4 linear terms, 4 quadratic terms, 6 interaction terms, 4 cubic terms, and 17 cubic interaction terms. actually the total number of terms generated by the *polynomialFeatures* class in scikit-learn can be calculated using the binomial coefficient with the general formulation of ((n+p)|p) with n=4 (for 4 input features) and p=3 as the degree of polynomial.

Among the various polynomial fits applied to pressure recovery and distortion data, the seventh-degree polynomial exhibits superior performance, as evidenced by Fig. 5. This performance is observed in both minimizing errors and maximizing residual values.

The potential for overfitting was carefully mitigated during the development of the 7th-order polynomial model. The dataset was divided into training and test subsets, and the model's performance was validated on the independent test data. Additionally, the complexity of the polynomial model was evaluated against lower-order polynomials, ensuring that the 7th-order model provided the best trade-off between accuracy and robustness. Normalization of input variables further enhanced the model's stability and reduced susceptibility to overfitting.

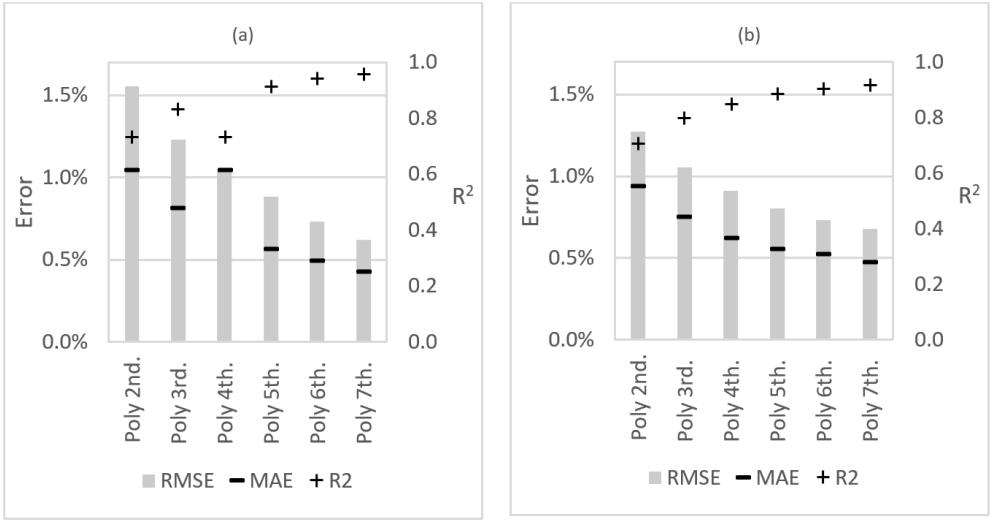


Fig. 5 Error and R2 values for different polynomial fits on (a) PR and (b) DC60 data

Thus, the seventh-degree polynomial (Poly 7th.) is chosen as the representative of polynomial models for subsequent comparisons with other modeling methods.

Preliminary evaluation of neural network models has been performed utilizing a grid search approach across a range of hyper parameter values. Different combinations of the number of hidden layers (L = 1, 2, and 3) and the number of nodes per layer (N = 8, 16, 32, 64, 128, 256, 512, and 1024) were examined. The activation function employed was systematically evaluated using three widely used choices: sigmoid, tanh, and Rectified Linear Unit (ReLU). This evaluation aimed to discern the impact of each activation function on predictive outcomes. As evaluation metrics, mean squared error (MSE) and mean absolute error (MAE) were employed, representing established benchmarks for assessing model accuracy. The analysis revealed that, among the considered activation functions, ReLU demonstrated superior efficacy in addressing the specific complexities of the problem at hand. This finding underscores the importance of judicious selection of the activation function and highlights ReLU as an optimal choice for enhancing the predictive capabilities of neural networks within the context of this study's problem domain.

In the context of neural network training, an epoch is one complete pass through the entire training dataset during the training phase. During each epoch, the neural network's parameters (weights and biases) are adjusted based on the error (loss) calculated on the training data. The loss is considered to be the mean squared error in the current study. In practice, the training dataset is divided into smaller subsets called batches. The model's parameters are updated after processing each batch. The size of the batch is a hyper parameter known as the batch size. One epoch is completed when the entire training dataset has been processed by the neural network once. It consists of multiple batches. An iteration refers to the process of updating the model's parameters using a single batch of data. So, for example in an epoch with a batch size of 32, there would be

$$\frac{\text{total number of samples}}{\text{Batch size}} = \frac{4000 \times 0.8}{32} = 100$$

Iterations.

If the epoch is selected to number “A”, the neural network processes the entire training dataset “A” times. The model's parameters are updated “A” times based on the error calculated on the training data. It goes through “A” cycles of forward and backward passes.

As it is mentioned, the number of epochs is a hyper parameter that is required to be tuned during the training process. Too few epochs may result in under-fitting, where the model hasn't learned the underlying patterns in the data. Too many epochs may lead to overfitting, where the model starts memorizing the training data and performs poorly on new, unseen data. During the training, the model's performance on a validation dataset is being monitored and the training is stopped when the performance starts to degrade, indicating that the model is overfitting. This is often done using early stopping technique. Typical effect of change of number of nodes in 1-hidden layer NN is shown in Fig. 6. The optimized number of epochs are obtained using the early stopping technique.

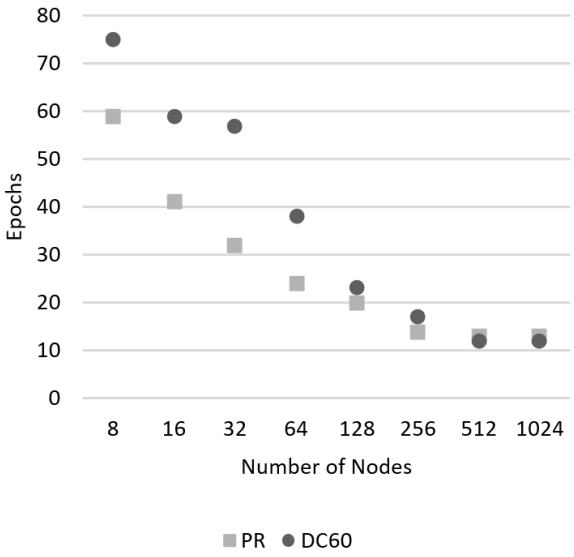


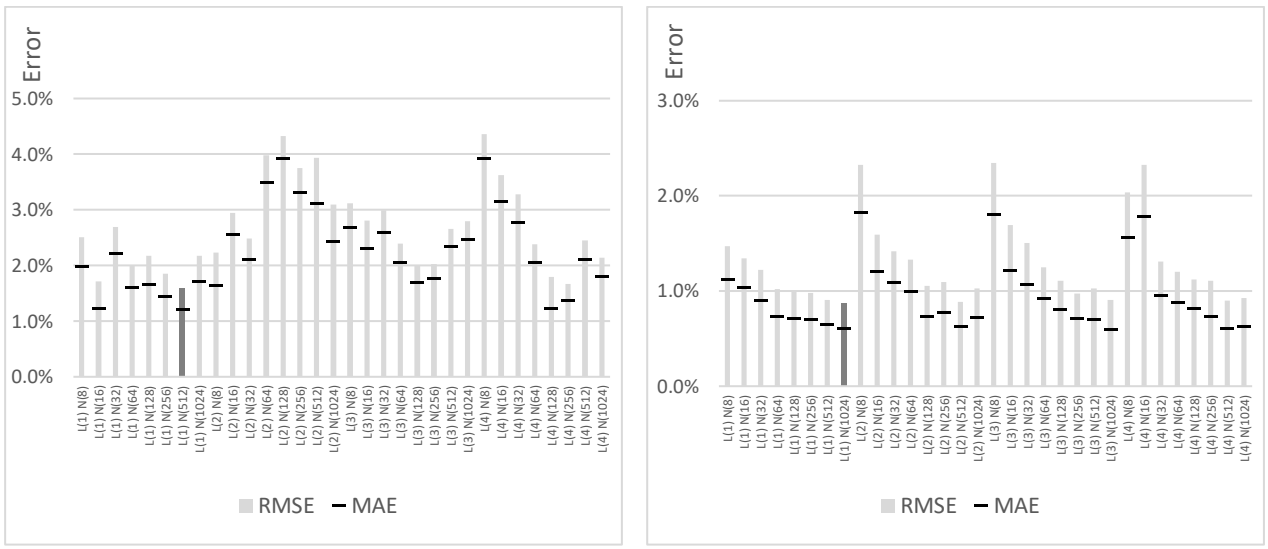
Fig. 6 Effect of number of nodes in a 1-layer NN on the number of epochs resulting minimum RMSE

Analysis of the RMSE and MAE results revealed that no single optimal neural network configuration emerged for concurrently modeling both PR and DC60. The preferred model (based on RMSE results) exhibited variability across different training runs. Figure 7 exemplifies a typical grid search outcome for both PR and DC60 values. In this specific instance, the optimal NN configuration for PR was identified as L(1) N(512), while for DC60 it was L(1) N(1024).

Analysis of the results revealed that, for the majority of training runs, the optimal model configurations for both PR and DC60 were found among the one-layer neural networks. Considering the significant computational efficiency offered by simpler networks, the study focused exclusively on one-layer networks to identify the optimal number of nodes. This optimization process involved 10 consecutive training runs, with the majority of cases converging on a configuration of 1024 nodes for both PR and DC60. Consequently, a one-layer neural network with 1024 nodes, denoted as "L(1) N(1024)", is designated as the representative model for subsequent meta-model comparisons.

While configurations with fewer nodes, such as L(1) N(16), showed similar error metrics in certain training runs, L(1) N(512) was chosen for its consistent performance and robustness across multiple iterations. The larger capacity of L(1) N(512) allowed it to better capture the complex relationships in the data, particularly for the distortion coefficient (DC60), while maintaining computational efficiency. This selection was based on achieving a balance between model complexity and predictive accuracy.

The neural network (NN) architecture was optimized based on a balance between accuracy and complexity.



(a) (b)

Fig. 7 Overall error (RMSE and MAE) for different NN models on PR (a) and DC60 (b) variables

While increasing the number of layers could potentially improve accuracy, preliminary experiments with deeper networks showed only marginal gains in performance metrics (e.g., RMSE and CCC), accompanied by increased risk of overfitting due to the limited dataset size. The chosen architecture (L(1) N(512)) provided a competitive balance, achieving robust performance with minimal risk of overfitting and low computational cost. Future studies could explore deeper architectures in conjunction with larger datasets or additional regularization techniques to further enhance accuracy.

In figures and charts presented throughout this study, the abbreviation "NN" refers specifically to the chosen "L(1) N(1024)" neural network configuration hereinafter.

Overfitting was mitigated through several measures, including an 80-20 train-test split, cross-validation during hyperparameter tuning, and regularization techniques. For the Random Forest Regressor (RFR), regularization was achieved by limiting tree depth (max_depth) and controlling the minimum number of samples for splits and leaf nodes (min_samples_split and min_samples_leaf). For neural networks, regularization methods such as early stopping, dropout (0.2), and weight decay (L2 regularization) were employed. Input normalization further reduced the risk of overfitting by ensuring consistent variable scaling. Final model selection prioritized performance on the independent test set to ensure generalization.

3. RESULTS AND DISCUSSION

The dataset used in this study was designed to capture a wide range of operating conditions for a fixed intake geometry. While it effectively represents the studied parameter space, it does not account for variations in geometry or extreme operating conditions. Consequently, the models, including Random Forest Regressor (RFR), may not generalize well to scenarios involving significant geometric changes. Future work could extend the dataset to include additional geometries and test the extrapolation capabilities of the models. This would provide insights into their robustness for more generalized applications. Obtained results for total pressure recovery and its distortion are given and explained in the following.

3.1 Modeling Total Pressure Recovery (PR)

Figure 8 presents a comparative assessment of the accuracy of various modeling techniques for predicting total pressure recovery (PR). It is worth to note that Figure 7-a presents RMSE values obtained during hyperparameter optimization using cross-validation on the training dataset. These values represent relative performance among various configurations but are not directly comparable to the final RMSE values reported in Figure 8, which correspond to the model's performance on the independent test dataset. The comparison in Figure 8 employs a seventh-order polynomial fit as the representative of polynomial-based models and a one-layer neural network with 1024 nodes as the representative of neural network-based models.

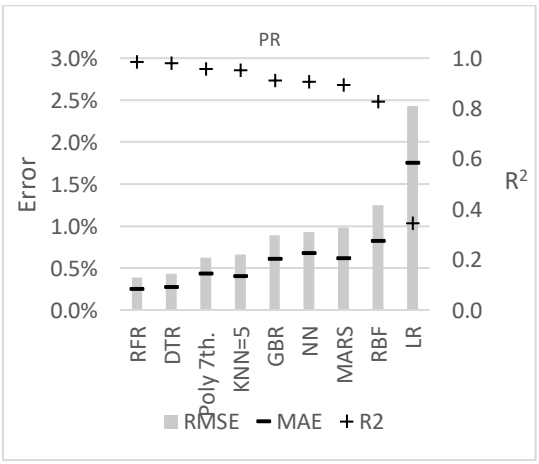


Fig. 8 Overall Error (RMSE and MAE) and R2 for different models on pressure recovery (PR) data

The 7th-order polynomial model demonstrated strong predictive accuracy on out-of-sample data, as indicated by its low RMSE and high CCC values on the test dataset. While the use of high-order polynomials necessitates careful handling to prevent overfitting, the observed consistency in error metrics between training and test data confirms the model's ability to generalize effectively within the studied parameter range.

The Random Forest Regressor (RFR) model was optimized using a grid search with cross-validation. The hyperparameters evaluated included the number of trees (*n_estimators*), maximum tree depth (*max_depth*), minimum samples required for a split (*min_samples_split*), minimum samples required at a leaf node (*min_samples_leaf*), and the maximum features considered for splitting (*max_features*). The optimal configuration was determined to be 200, 20, 5, 2, and 'sqrt', respectively. This configuration achieved the lowest RMSE on the validation dataset.

Among the evaluated models, Random Forest Regressor (RFR) emerges as the most promising, exhibiting a very low overall error and a high R2 value. Decision Tree Regressor (DTR) and seventh-order polynomials rank second and third, respectively, in terms of overall RMSE. Conversely, simple models like linear regression demonstrate significantly low accuracy.

Figure 9 presents the prediction intervals for different models, ranked in ascending order of RMSE based on the findings of Figure 8. A narrow prediction interval width for RFR and DTR signifies greater confidence in their predictions, implying that the predicted values are likely to fall within a smaller range. The overlap between the prediction intervals of these two models further suggests that they generate similar predictions with comparable uncertainty levels. Polynomial and KNN models also exhibit acceptable prediction intervals. Conversely, the wide prediction intervals associated with RBF and LR models indicate higher uncertainty and reduced precision in their predictions. The dashed 45-degree line serves as an indicator of the deviation between predicted and true values, and consequently, it can also be interpreted as a representation of residuals.

The over-prediction observed at lower values of the target variable in Figure 9 likely results from a combination of data imbalance and inherent biases in the models. The dataset contains fewer samples in the low-value range, which may lead to reduced generalization in this region. To mitigate this bias in future work, resampling or weighting techniques could be applied to ensure better representation of lower values. Additionally, post-training calibration methods and custom loss functions may be explored to correct systematic prediction biases.

Figure 10 (a) presents various correlation coefficients calculated for the PR models, ranked in descending order of CCC. Notably, the ranking based on CCC values differs from the ranking based on RMSE shown in Figure 8. As previously noted, CCC is useful for reliability analysis. While most models exhibit satisfactory concordance, the agreement between predicted and actual PR values is strongest for RFR, DTR, and KNN=5 models. This finding also suggests potential robustness in these models, provided that their bias results are promising. By examining the SCC and KRCC values, a strong monotonic relationship between predicted and actual data is observed for RFR, DTR, and KNN=5 models.

Figure 10 (b) illustrates the overall bias found in different models. Bias is defined as the difference between the average of predicted values and the average of test values. A positive bias indicates overestimation of the majority of predicted values, while a negative bias suggests underestimation.

The bias trends observed in Figure 10 reveal systematic over-prediction in the Pressure Recovery (PR) model at lower values and under-prediction in the Distortion Coefficient (DC60) model at mid-to-high values. These trends likely result from the sparsity of data in extreme value regions and the smoothing effects of models such as Random Forest Regressor (RFR) and Neural Networks (NN). Addressing these biases in future studies could involve increasing data density in underrepresented regions, applying post-training calibration techniques, or employing hybrid models to better capture complex relationships in the data.

Once again, RFR exhibits the best performance in terms of bias, with a slight overestimation observed. The majority of models tend to overestimate the PR data, with only polynomial and RBF models showing a slight underestimation.

3.2 Modeling Pressure Distortion Coefficient (DC60)

Figure 11 presents a comparative analysis of the accuracy of nine existing models for the second dependent variable, total pressure distortion coefficient (DC60). Employing Root Mean Squared Error (RMSE) as the primary metric, Random Forest Regressor (RFR) emerges as the most promising model, exhibiting exceptionally low overall error and a highly favorable R2 value. K-Nearest Neighbor (KNN) with K=5 and 7th-order polynomial follow in second and third positions based on overall RMSE. This comparison underscores the detrimental impact of simplifying the model to a linear nature, as evidenced by the reduction in both accuracy (increased error) and variability (lower R2).

The Gradient Boosting Regression (GBR) model exhibited suboptimal performance compared to the Random Forest Regressor (RFR), as shown in Fig. 11. This is likely due to GBR's sensitivity to overfitting, particularly in sparse data regions, where its sequential training approach can amplify noise. While hyperparameter tuning was conducted, GBR's performance depends strongly on parameters such as *learning_rate* and *n_estimators*, which may not have been fully optimized for this dataset. In contrast, RFR's ensemble averaging reduces overfitting risks and enhances robustness to data heterogeneity, leading to superior performance in this study.

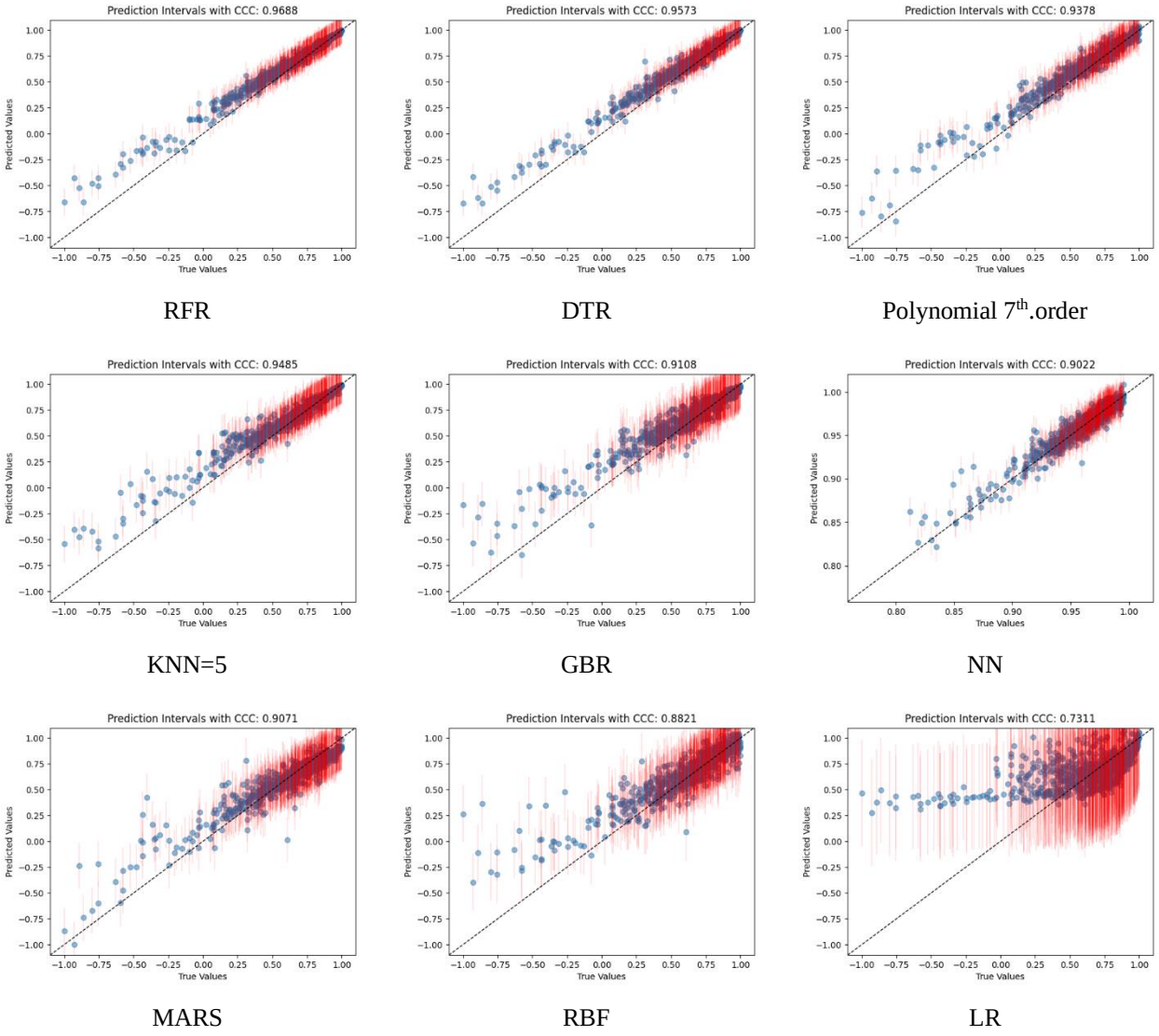
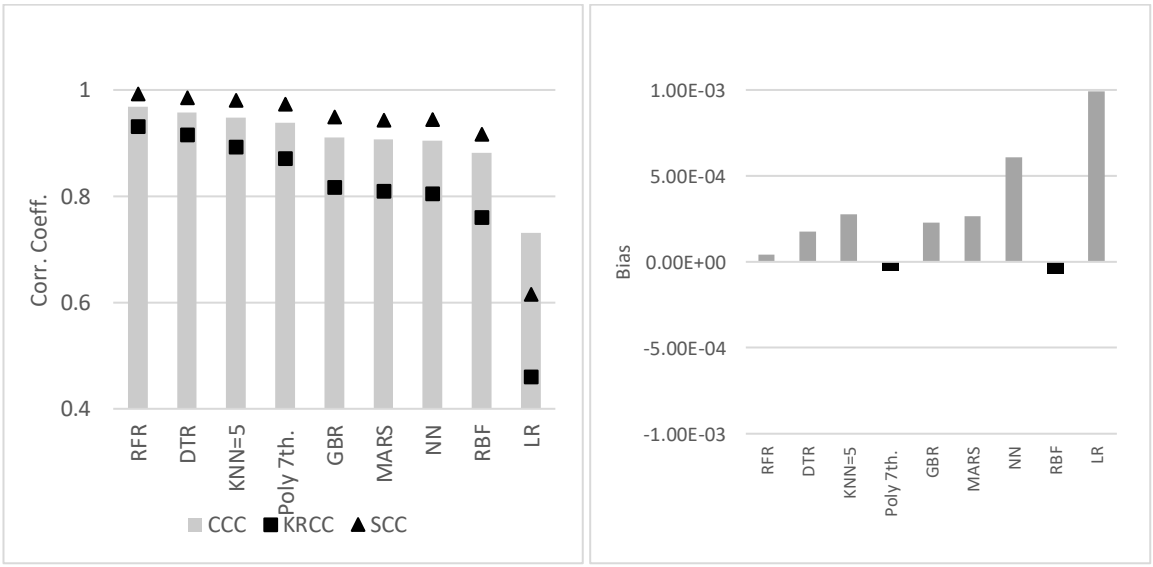


Fig. 9 Prediction intervals for different models of coded PR data. Vertical red lines indicate uncertainty of prediction. Concordance Correlation Coefficient (CCC) is also given for each model



(a) (b)

Fig. 10 Concordance Correlation Coefficient (CCC), Kendall Tau (KRCC), and Spearman correlation coefficient (SCC) for different models on PR data (a) Overall bias in different models (b)

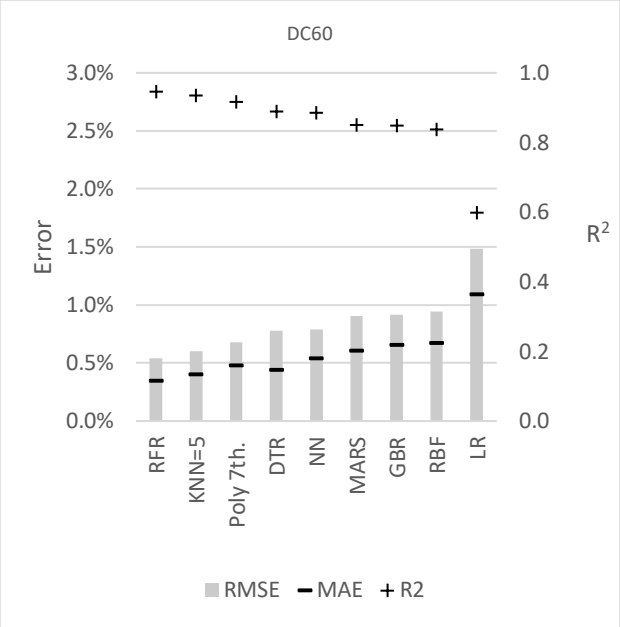


Fig. 11 Overall Error (RMSE and MAE) and R2 for different models on pressure distortion coefficient (DC60) data

Figure 12 presents the prediction intervals for various models evaluated on the DC60 data. The individual figures are arranged in ascending order of Root Mean Squared Error (RMSE), adhering to the ranking established in Fig. 11. Notably, Figure 12 reveals that Random Forest Regressor (RFR) exhibits the narrowest prediction intervals, signifying the highest level of confidence among the nine models in predicting DC60. K-Nearest Neighbor (KNN) with K=5 and 7th-order polynomials demonstrate comparable levels of uncertainty. Additionally, the trend observed in the Linear Regression (LR) graph reiterates the non-linear nature of the data and the inadequacy of this model for accurate prediction. Consistent with the analysis of Pressure Ratio (PR) data, it is crucial to consider other metrics alongside prediction intervals for a comprehensive evaluation of model performance.

The consistent under-prediction observed at higher DC60 values in Fig. 12 likely stems from sparse data representation in this range and the smoothing effects of the models. Random Forest Regressor (RFR) and Neural Networks (NN) prioritize overall accuracy, often at the expense of extreme-value prediction. Polynomial models, while effective for general trends, struggle with extrapolation in high-value regions. Future work could address this limitation through targeted data augmentation, custom loss functions to prioritize high-value accuracy, and post-training calibration techniques to adjust model predictions.

Figure 13(a) further corroborates the findings by ranking the models based on their Concordance Correlation Coefficient (CCC) values. Notably, RFR once again establishes itself as the leading model, while Linear Regression (LR) demonstrably presents the lowest level of concordance. Also presented in the figure the Spearman correlation coefficient (SCC) and Kendall's Tau (KRCC) for various models. Notably, the models exhibit a consistent ranking across both criteria. In terms of performance, Random Forest Regressor (RFR) and KNN with K=5 emerge as the frontrunners. Furthermore, it is noteworthy that the desired parametric polynomial model also demonstrates satisfactory performance on the DC60 data.

Figure 13(b) reveals that despite the relatively small absolute value of bias error, most models underestimate the predicted DC60 values. This observation stands in contrast to the tendency of most meta-models to overestimate the PR variable, as illustrated in Figure 10(b). Notably, Decision Tree Regression (DTR) exhibits the smallest bias error for DC60, while Neural Networks (NN) produce the largest bias. However, the training time for NNs was twice that required by other models.

The 7th-degree polynomial model demonstrated strong predictive capabilities, achieving a CCC of 0.9378 and an RMSE comparable to more complex machine learning models. While Random Forest Regressor (RFR) displayed marginally better overall performance, the 7th-degree polynomial emerged as a competitive and interpretable parametric alternative, especially in applications favoring explicit mathematical representation of the relationships.

4. CONCLUSION

This study conducted a comprehensive investigation into the performance of various surrogate (or meta) modeling techniques for predicting two key aerodynamic parameters in an air intake system: total pressure recovery (PR) and total pressure distortion coefficient (DC60). A diverse range of models, encompassing both traditional and advanced machine learning algorithms, were evaluated using a rigorous set of metrics, including accuracy, reliability, robustness, bias, and computational efficiency. Findings of this study are pointed below.

The study identified Random Forest Regressor (RFR) as the most effective model for PR prediction, outperforming others across all metrics considered, including RMSE, R2, CCC, SCC, KRCC, and bias. While RMSE provided valuable insight into overall accuracy, relying solely on this metric proved insufficient.

RFR exhibited marginal improvement in performance in terms of bias and concordance metrics, emphasizing the need for multi-faceted evaluation. Simple models like linear regression demonstrated limited effectiveness for PR prediction due to their inability to capture the non-linear relationships in the data. In contrast, the 7th-degree polynomial, a higher-order parametric approach, achieved competitive results, including a CCC of 0.9378, low RMSE (~0.5%), and minimal bias. These results highlight the suitability of higher-order parametric models for certain applications, even when compared with more advanced machine learning techniques. The majority of models exhibited a tendency to overestimate PR values, suggesting the need for further investigation and potential bias mitigation strategies.

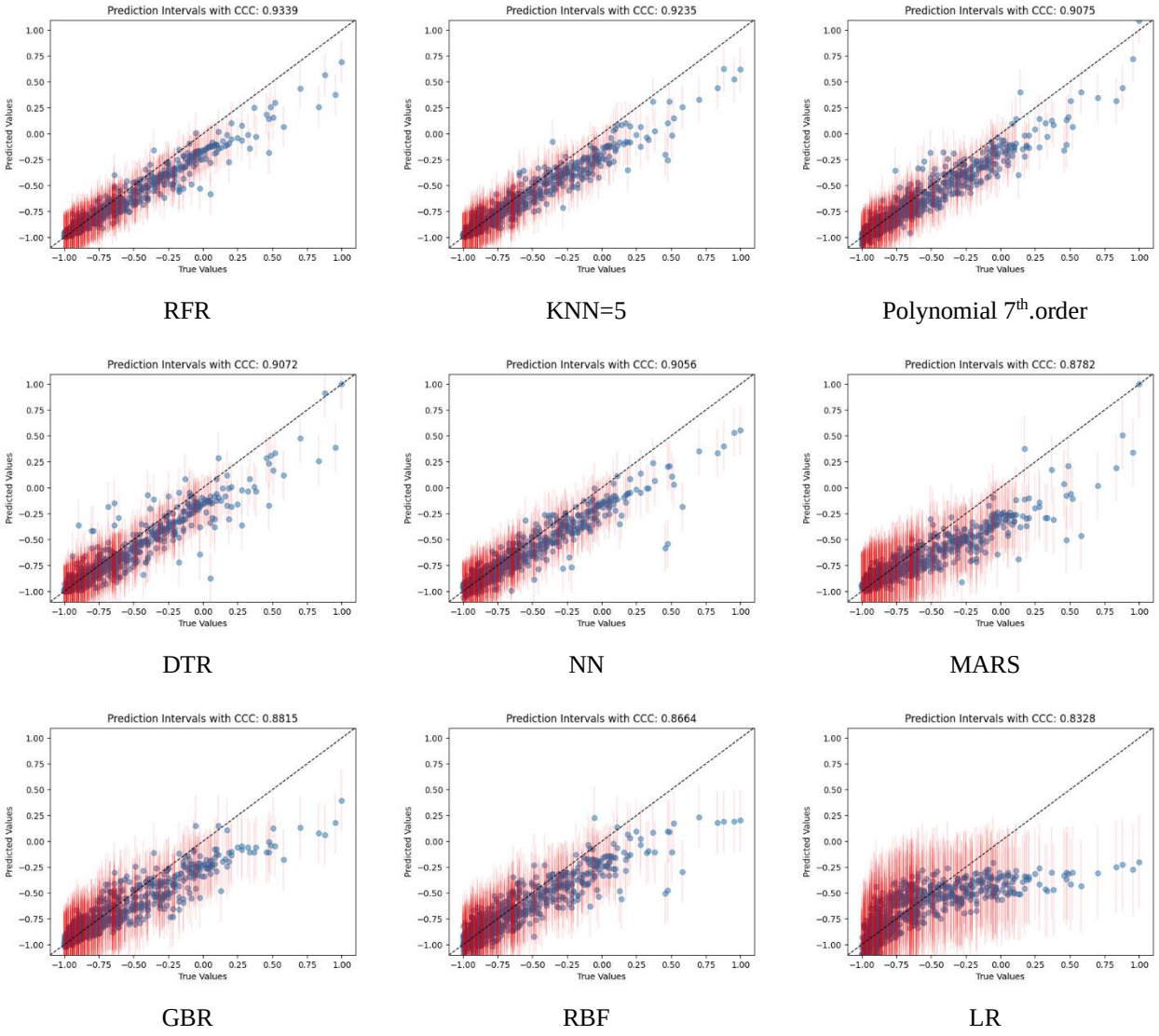
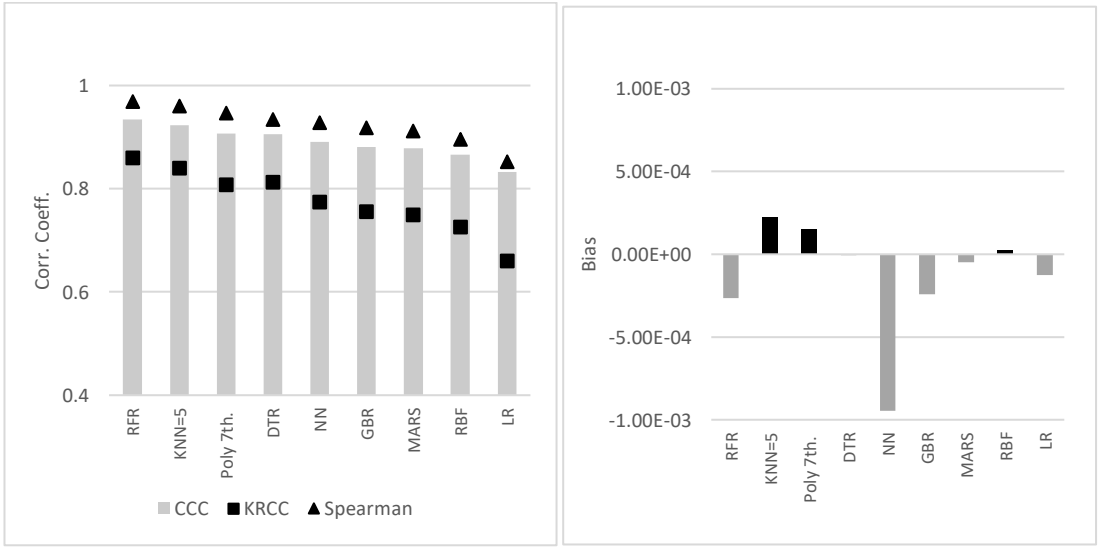


Fig. 12 Prediction intervals for different models of coded DC60 data. Vertical red lines indicate uncertainty of prediction. Concordance Correlation Coefficient (CCC) is also given for each model



(a) (b)

Fig. 13 Concordance Correlation Coefficient (CCC), Kendall Tau (KRCC), and Spearman correlation coefficient (SCC) for different models on DC60 data (a) Overall bias in different models (b)

RFR maintained its marginal leadership for DC60 prediction, showcasing exceptionally low error and high R2, underscoring its versatility. KNN with K=5 and 7th-order polynomials emerged as promising alternatives, demonstrating satisfactory performance in terms of accuracy and uncertainty quantification. As with PR, linear regression proved inadequate for DC60 prediction due to its inability to capture the complex non-linear relationships within the data. Contrary to the overestimation observed in PR, most models underestimated DC60 values, highlighting the need for model-specific bias analysis. While Neural Networks displayed promising DC60 prediction capabilities, their significantly longer training time compared to other models necessitates careful consideration. The 7th-order polynomial model demonstrated superior performance compared to other parametric models, showcasing the effectiveness of carefully chosen polynomial models for specific applications.

Analyzing findings for both PR and DC60 reveals that RFR exhibits remarkable performance across both parameters, establishing itself as a versatile and robust choice. Model selection should be based on comprehensive evaluation utilizing multiple metrics, avoiding reliance solely on accuracy measures. Traditional models like linear regression have limited applicability in complex engineering domains where non-linear relationships dominate. Understanding and addressing bias tendencies for individual models is crucial for ensuring reliable predictions. Computational efficiency remains a critical factor, requiring a balance between model performance and training time.

Future research efforts should focus on exploring alternative advanced machine learning algorithms for further performance enhancements, incorporating uncertainty quantification techniques to assess prediction confidence intervals, investigating the impact of feature engineering and selection on model performance, exploring ensemble learning and transfer learning approaches for improved generalizability and performance across different domains, and adapting existing models to incorporate domain-specific knowledge and constraints for even more accurate and reliable predictions.

ACKNOWLEDGEMENTS

The authors would like to acknowledge Turkish Aerospace Industries (TUSAŞ) for providing the necessary resources including access to databases.

CONFLICT OF INTEREST

Authors declare that they have no conflicts to disclose.

AUTHORS CONTRIBUTION

Human Amiri: Conceptualization, methodology, formal analysis, writing, and editing of the original draft. **Umut Can Kucuk**: Analysis, Manuscript revision. **Onur Kucukoglu:** Experimental campaign. **Yigit Firat Kuscu:** Experimental campaign. **Osman Veysel Ozdemir:** Manuscript revision.

DECLARATION OF GENERATIVE AI AND AI-ASSISTED TECHNOLOGIES IN THE WRITING PROCESS

During the preparation of this work, the author(s) engaged the assistance of ChatGPT, an AI language model developed by OpenAI, to enhance the clarity and correctness of the content. ChatGPT was employed for grammatical correction and refinement of the writing. After using this tool/service, the author(s) reviewed and edited the content as needed and take(s) full responsibility for the content of the publication.

REFERENCES

Altman, N. S. (1992). An introduction to kernel and nearest-neighbor nonparametric regression. *The American Statistician, 46*(3), 175. https://doi.org/10.2307/2685209

Askari, R., & Soltani, M. R. (2020). Flow asymmetry in a Y-Shaped diverterless supersonic inlet: A novel finding. *AIAA Journal, 58*(6), 2609–2620. https://doi.org/10.2514/1.J059006

Aziz, N., Akhir, E. A. P., Aziz, I. A., Jaafar, J., Hasan, M. H., & Abas, A. N. C. (2020). *A study on gradient boosting algorithms for development of AI monitoring and prediction systems.* 2020 International Conference on Computational Intelligence (ICCI), 11–16. https://doi.org/10.1109/ICCI51257.2020.9247843

Box, G. E. P., & Draper, N. R. (1987). *Empirical model-building and response surfaces.* Wiley.

Chen, V. C. P., Tsui, K. L., Barton, R. R., & Meckesheimer, M. (2006). A review on design, modeling and applications of computer experiments. *IIE Transactions, 38*(4), 273–291. https://doi.org/10.1080/07408170500232495

Drężek, P. S., Kubacki, S., & Żółtak, J. (2022). Multi-objective surrogate model-based optimization of a small aircraft engine air-intake duct. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering, 236*(14), 2909–2921. https://doi.org/10.1177/09544100211070868

Erickson, C. B. (2019). Adaptive *computer experiments for metamodeling.* Northwestern University.

El-Sayed, A. F. (2016). Aero-engines intake: A review and case study. *Journal of Robotics and Mechanical Engineering Research, 1*(3), 35–42. https://doi.org/10.24218/jrmer.2016.15

Forrester, A. I. J., Sóbester, A., & Keane, A. J. (2008). *Engineering design via surrogate modelling: A practical guide* (1st ed.). Wiley. https://doi.org/10.1002/9780470770801

Friedman, J. H. (1991). Multivariate adaptive regression splines. *The Annals of Statistics, 19*(1).

https://doi.org/10.1214/aos/1176347963

Friedman, L. W., & Pressman, I. (1988). The metamodel in simulation analysis: can it be trusted? *The Journal of the Operational Research Society, 39*(10), 939. https://doi.org/10.2307/2583045

Gabriel Pereira Gouveia da Silva, F. M. C. (2019). *Neural network metamodeling for aerodynamic optimization efficiency improvement.* IV Simpósio do Programa de Pós-Graduação em Engenharia Mecânica da EESC-USP (SiPGEM/EESC-USP).

Jena, M., & Dehuri, S. (2020). Decision tree for classification and regression: a state-of-the art review. *Informatica, 44*(4). https://doi.org/10.31449/inf.v44i4.3023

Kianifar, M. R., & Campean, F. (2020). Performance evaluation of metamodelling methods for engineering problems: Towards a practitioner guide. *Structural and Multidisciplinary Optimization, 61*(1), 159–186. https://doi.org/10.1007/s00158-019-02352-1

Kim, H., Loh, W. Y., Shih, Y. S., & Chaudhuri, P. (2007). Visualizable and interpretable regression models with good prediction power. *IIE Transactions, 39*(6), 565–579. https://doi.org/10.1080/07408170600897502

Kleijnen, J. P. C. (2009). Kriging metamodeling in simulation: A review. *European Journal of Operational Research, 192*(3), 707–716. https://doi.org/10.1016/j.ejor.2007.10.013

Kleijnen, J. P. C. (2017). Regression and kriging metamodels with their experimental designs in simulation: A review. *European Journal of Operational Research, 256*(1), 1–16. https://doi.org/10.1016/j.ejor.2016.06.041

Kyprioti, A. P., Zhang, J., & Taflanidis, A. A. (2020). Adaptive design of experiments for global Kriging metamodeling through cross-validation information. *Structural and Multidisciplinary Optimization, 62*(3), 1135–1157. https://doi.org/10.1007/s00158-020-02543-1

Le Clainche, S., Ferrer, E., Gibson, S., Cross, E., Parente, A., & Vinuesa, R. (2023). Improving aircraft performance using machine learning: A review. *Aerospace Science and Technology, 138*, 108354. https://doi.org/10.1016/j.ast.2023.108354

Li, J., Du, X., & Martins, J. R. R. A. (2022). Machine learning in aerodynamic shape optimization. *Progress in Aerospace Sciences, 134*, 100849. https://doi.org/10.1016/j.paerosci.2022.100849

Li, Y. F., Ng, S. H., Xie, M., & Goh, T. N. (2010). A systematic comparison of metamodeling techniques for simulation optimization in decision support systems. *Applied Soft Computing, 10*(4), 1257–1273. https://doi.org/10.1016/j.asoc.2009.11.034

Morris, M. D., & Mitchell, T. J. (1995). Exploratory designs for computational experiments. *Journal of Statistical Planning and Inference, 43*(3), 381–402. https://doi.org/10.1016/0378-3758(94)00035-T

Patel, T., Singh, S. N., & Seshadri, V. (2005). Characteristics of Y-Shaped rectangular diffusing duct at different inflow conditions. *Journal of Aircraft, 42*(1), 113–120. https://doi.org/10.2514/1.4690

Pearson, K. (1895). Note on regression and inheritance in the case of two parents. *Proceedings of the Royal Society of London Series I, 58*, 240–242.

Poggi, C., Rossetti, M., Serafini, J., Bernardini, G., Gennaretti, M., & Iemma, U. (2022). Neural network meta–modelling for an efficient prediction of propeller array acoustic signature. *Aerospace Science and Technology, 130*, 107910. https://doi.org/10.1016/j.ast.2022.107910

Schneider, A., Hommel, G., & Blettner, M. (2010). Linear regression analysis. *Deutsches Ärzteblatt International.* https://doi.org/10.3238/arztebl.2010.0776

Seddon, J., & Goldsmith, E. L. (1999). Intake Aerodynamics. 2nd ed., AIAA Education Series, American Institute of Aeronautics and Astronautics.

Simpson, T. W., Poplinski, J. D., Koch, P. N., & Allen, J. K. (2001). Metamodels for computer-based engineering design: Survey and recommendations. *Engineering with Computers, 17*(2), 129–150. https://doi.org/10.1007/PL00007198

Singh, U., Rizwan, M., Alaraj, M., & Alsaidan, I. (2021). A machine learning-based gradient boosting regression approach for wind power production forecasting: A step towards smart grid environments. *Energies, 14*(16), 5196. https://doi.org/10.3390/en14165196

Sun, G., & Wang, S. (2019). A review of the artificial neural network surrogate modeling in aerodynamic design. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering, 233*(16), 5863–5872. https://doi.org/10.1177/0954410019864485

Van Gelder, L., Das, P., Janssen, H., & Roels, S. (2014). Comparative study of metamodelling techniques in building energy simulation: Guidelines for practitioners. *Simulation Modelling Practice and Theory, 49*, 245–257. https://doi.org/10.1016/j.simpat.2014.10.004

Wang, H., Shan, S., Wang, G. G., & Li, G. (2011). Integrating least square support vector regression and mode pursuing sampling optimization for crashworthiness design. *Journal of Mechanical Design, 133*(4), 041002. https://doi.org/10.1115/1.4003840

Wang, L., Misra, G., & Bai, X. (2019). A K nearest neighborhood-based wind estimation for rotary-wing VTOL UAVs. *Drones, 3*(2), 31. https://doi.org/10.3390/drones3020031

Wu, H., Zhao, Y. P., & Hui-Jun, T. (2022). A hybrid of fast K-nearest neighbor and improved directed

acyclic graph support vector machine for large-scale supersonic inlet flow pattern recognition. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, *236*(1), 109–122. https://doi.org/10.1177/09544100211008601

Yan, Q., Wan, Z., & Yang, C. (2023). Flight load calculation using neural network residual kriging. *Aerospace*, *10*(7), 599. https://doi.org/10.3390/aerospace10070599

Yang, R. J., Gu, L., Liaw, L., Gearhart, C., Tho, C. H., Liu, X., & Wang, B. P. (2000). *Approximations for safety optimization of large systems.* 26th Design Automation Conference, 763–772. https://doi.org/10.1115/DETC2000/DAC-14245

Yang, T., Zhiyong, L., Neng, X., Yan, S., & Jun, L. (2018). Optimization of positional parameters of close-formation flight for blended-wing-body configuration. *Heliyon*, *4*(12), e01019. https://doi.org/10.1016/j.heliyon.2018.e01019

Zan, B. W., Han, Z. H., Xu, C. Z., Liu, M. Q., & Wang, W. Z. (2022). High-dimensional aerodynamic data modeling using a machine learning method based on a convolutional neural network. *Advances in Aerodynamics*, *4*(1), 39. https://doi.org/10.1186/s42774-022-00128-8