Les Cahiers du GERAD ISSN: 0711–2440

Transfer learning in surrogate modeling with emphasis on aircraft design

A. Tfaily, N. Bartoli, Y. Diouane, M. Kokkolaras

G–2025–50

July 2025

Citation suggérée : A. Tfaily, N. Bartoli, Y. Diouane, M. Kokkolaras (July 2025). Transfer learning in surrogate modeling with emphasis on aircraft design, Rapport technique, Les Cahiers du GERAD G– 2025–50, GERAD, HEC Montréal, Canada.

Avant de citer ce rapport technique, veuillez visiter notre site Web (https://www.gerad.ca/fr/papers/G-2025-50) afin de mettre à jour vos données de référence, s'il a été publié dans une revue scientifique.

Suggested citation: A. Tfaily, N. Bartoli, Y. Diouane, M. Kokkolaras (July 2025). Transfer learning in surrogate modeling with emphasis on aircraft design, Technical report, Les Cahiers du GERAD G–2025–50, GERAD, HEC Montréal, Canada.

Before citing this technical report, please visit our website (https://www.gerad.ca/en/papers/G-2025-50) to update your reference data, if it has been published in a scientific journal.

La publication de ces rapports de recherche est rendue possible grâce au soutien de HEC Montréal, Polytechnique Montréal, Université McGill, Université du Québec à Montréal, ainsi que du Fonds de recherche du Québec – Nature et technologies.

Dépôt légal – Bibliothèque et Archives nationales du Québec, 2025
– Bibliothèque et Archives Canada, 2025

The publication of these research reports is made possible thanks to the support of HEC Montréal, Polytechnique Montréal, McGill University, Université du Québec à Montréal, as well as the Fonds de recherche du Québec – Nature et technologies.

Legal deposit – Bibliothèque et Archives nationales du Québec, 2025
– Library and Archives Canada, 2025

GERAD HEC Montréal
3000, chemin de la Côte-Sainte-Catherine
Montréal (Québec) Canada H3T 2A7

Tél. : 514 340-6053
Téléc. : 514 340-5665
info@gerad.ca
www.gerad.ca

Transfer learning in surrogate modeling with emphasis on aircraft design

Ali Tfaily [a, b, e]

Nathalie Bartoli [d]

Youssef Diouane [c, e]

Michael Kokkolaras [a, e]

[a] *Department of Mechanical Engineering McGill University, Montréal (Qc), Canada*

[b] *Advanced Product Development, Bombardier, Montréal (Qc), Canada*

[c] *Department of Mathematical and Industrial Engineering, Polytechnique Montréal, Montréal (Qc), Canada*

[d] *ONERA/DTIS, Université de Toulouse, Toulouse, France*

[e] *GERAD, Montréal (Qc), Canada*

ali.tfaily@mail.mcgill.ca
youssef.diouane@polymtl.ca
nathalie.bartoli@onera.fr
michael.kokkolaras@mcgill.ca

July 2025
Les Cahiers du GERAD
G–2025–50

Abstract : Surrogate modeling with insufficient data can lead to high prediction uncertainty and errors. A promising remedy to address this issue is the use of transfer learning techniques that leverage models built using data from other problems that are implicitly related to the problem of interest. We present an algorithm that uses transfer learning and mixtures of experts across different design space regions to improve the predictive capability of surrogate models. The algorithm uses existing data to divide a problem's design space into clusters and build ensembles of surrogate models in each cluster using a multi-criteria weighting method. The proposed algorithm is shown to be both accurate and flexible, allowing for automated transfer learning with tuning parameters that cater for different problem types. The multi-criteria approach enables transfer learning in constrained Bayesian optimization by weighing models based on their shape, accuracy, and variance. The proposed method is demonstrated using aircraft conceptual design examples and showed up to 10% reduction in prediction errors.

Keywords: Transfer learning, surrogate modeling, ensemble of surrogates, mixture of experts, aircraft design

---

Acknowledgements: The authors would like to thank Jasveer Singh from Bombardier for his insights and support in setting up the industrial aircraft conceptual design problems.

1 Introduction

Engineers and designers typically face problems that require the development of surrogate models that emulate the behavior of computationally expensive functions. For aircraft design problems, typical uses of surrogate modeling include the simulation of certain disciplines such as aerodynamics, structures, weight estimation, and aircraft performance. In addition, aircraft designers use surrogate-based optimization algorithms to reduce the computational cost of aircraft design optimization problems [43] or to aid in solving problems with hierarchical and mixed-discrete variables such as system architecture optimization [8, 21, 48]. The ideal scenario in surrogate modeling is that there are enough training data to build accurate models. Collecting sufficient training data can be expensive, time-consuming, or even unrealistic.

Transfer learning is a sub-field of machine learning based on the idea of using information from one domain to another different but related domain. For example, information can be transferred from an existing machine learning model that classifies food images, built using existing data, to build a model that classifies plant images using new data. In this work, we refer to existing data as source data and data built based on the problem of interest as target data. Consequently, source models are built using source data and target models are built using target data. Transfer learning of surrogate models entails the use of information from source models to aid when there is insufficient target data to build reliable target models, or when trying to reduce the computational costs of building the target models [57, 67].

In this work, we consider transfer learning in surrogate modeling for aircraft design. Aircraft manufacturers rely heavily on models to simulate and improve an aircraft's design [11, 24, 42, 54]. A challenge faced by aircraft designers is the high prediction uncertainty and errors faced when creating surrogate models of certain disciplines with limited data [19]. Another challenge is the computational costs and time associated with expensive simulations and optimization in aircraft design [46]. We investigate the use of transfer learning for aircraft design and optimization to address the challenges above. Brunton *et al.* provide a review of the use of machine learning in aerospace applications [6] and highlight the need for transfer learning to aid in aircraft model-based engineering. Min *et al.* propose the use of transfer learning in the design of different aircraft engines [39]. They use source data from a different engine type relative to the target problem and add the source data to the target data to create individual target surrogate models. Tong *et al.* explore the application of transfer learning on aircraft engine design surrogate modeling by developing tools to predict turbo-fan core size using a database of two hundred engines [56]. Min *et al.* propose a method to transfer knowledge between source and target problems and apply their method to aircraft engine designs [38].

We are interested in Gaussian process surrogate models because of their flexibility and ability to quantify uncertainty to enable Bayesian optimization [58]. Gaussian processes are a type of surrogate model defined by their mean and covariance functions [58]. We introduce the use of Gaussian processes in aircraft design, which are prevalent in many disciplines and optimization applications. In [9], the authors used Gaussian process regression to determine both a mean estimate of the takeoff weight of an aircraft and the associated prediction interval, using observed data from the takeoff ground roll. The authors in [34] constructed surrogate models for airfoil design using multi-output Gaussian process regression models to predict airfoil lift, drag, and pitching-moment coefficients. In the field of aircraft design optimization, Gaussian processes are used in several applications. A conceptual design Bayesian optimization of business aircraft is presented in [43]. A wing aerodynamic design optimization using Gaussian processes is proposed in [44]. The authors of [55] used a co-Kriging method, a type of multi-output Gaussian process, to perform a multipoint drag minimization. However, the use of individual Gaussian processes in the context of transfer learning is not ideal due to the heterogeneity of data between different problems and due to the potential increased computational costs associated with increased amounts of data. Instead, we explore the use of ensembles of surrogate models for transfer learning.

The idea of using an ensemble of surrogates as a weighted average of individual surrogate models is proposed in [22]. The authors claim that using an ensemble of surrogates, which can be constructed without a significant expense compared to the cost of acquiring data, can prove effective in distilling correct trends from the data and improve robustness over the use of individual surrogate models. In [62], the authors demonstrated the use of ensembles of several types of surrogate models, including Gaussian processes, on a set of problems showing improvement in the robustness of results. Using ensembles of surrogates in transfer learning leverages existing data from source problems to combine source and target models with the goal of improving approximation performance and reducing training computational costs. The authors in [36] proposed an ensemble approach to transfer data between a set of source problems to a single target problem. The method entails the use of maximum a posteriori elaboration on a logistic regression model. In [14], the authors proposed a so-called multiple kernel learning framework for transfer learning where they apply a linear combination of multiple predefined kernels. In [40], the authors presented a framework for transfer learning based on modular variational Gaussian processes. The framework relies on a dictionary of pre-existing Gaussian processes that are used to complement a Gaussian process of a target task. The concept of ensemble of surrogates is also used in transfer learning of Bayesian optimization in [16, 17, 59, 60] where a weighted-average of models consisting of source surrogate models and target models is used.

This paper proposes a methodology for performing transfer learning between a set of source surrogate models and a target surrogate model, using an ensemble of surrogates defined over a partitioned design space. The main contributions of this paper are:

1. The use of posterior information to efficiently adapt source surrogate models to the target problem.
2. The construction of clusters within the design space of the target problem and the development of a mixture of experts in each cluster.
3. A multi-criteria weighting strategy to build the surrogate ensemble based on shape similarity, predictive accuracy, and variance.

This paper is organized as follows. In Section 2, we outline the aircraft design motivations and objectives that drive the need for transfer learning using ensembles of surrogate models, and we review related work that performs transfer learning using methods other than ensembles of surrogates. In Section 3, we propose the methodologies to develop ensembles of surrogate models using posterior information of source surrogate models, a mixture of experts that divides the design space, and a multi-criteria weighting method to combine the ensembles of surrogates. We then demonstrate the methodologies using analytical and aircraft design problems in Section 4. In Section 5, we close the paper by concluding on the methods presented and the claimed results.

2 Aircraft design surrogate modeling

This work is motivated by scenarios when aircraft designers have limited data from a specific target discipline and abundant data from another, i.e., source, discipline. The target disciplines could be complex to model and computationally expensive, or they could be not part of the core business of the designers' company and typically are obtained from external entities. In such scenarios we assume that we can transfer knowledge from other source disciplines implicitly assuming that these disciplines are relevant to the target discipline and therefore can be used to derive more accurate surrogate models or to obtain sensitivities to key input variables. In addition, we are interested in performing transfer learning for Bayesian optimization of aircraft to enable the use of existing (source) optimization data while performing a new target optimization. Finally, we are interested in performing transfer learning between different aircraft configurations, for example, we intend to leverage knowledge from available source data from a high-speed and high-sweep aircraft wing when designing a low-speed and low-sweep aircraft wing.

Of particular note in the aircraft design surrogate modeling scenarios is the relationship between source and target problems in specific areas of the design space. We discuss the example of the high-speed and low-speed wings. The lift, drag, and pitching moment of an aircraft operating at a fixed angle of attack change rapidly as the aircraft approaches the speed of sound. This change is called compressibility effect [23]. Its onset begins at a specific airspeed called the critical Mach number. It is driven by the formation of a normal shock that separates the flow on the aft part of an airfoil. An airplane with low-speed wings and thick airfoils will experience it as early as Mach 0.6, whereas a high-speed wing begins to experience it above Mach 0.80 as shown in Figure 1. The Mach number at

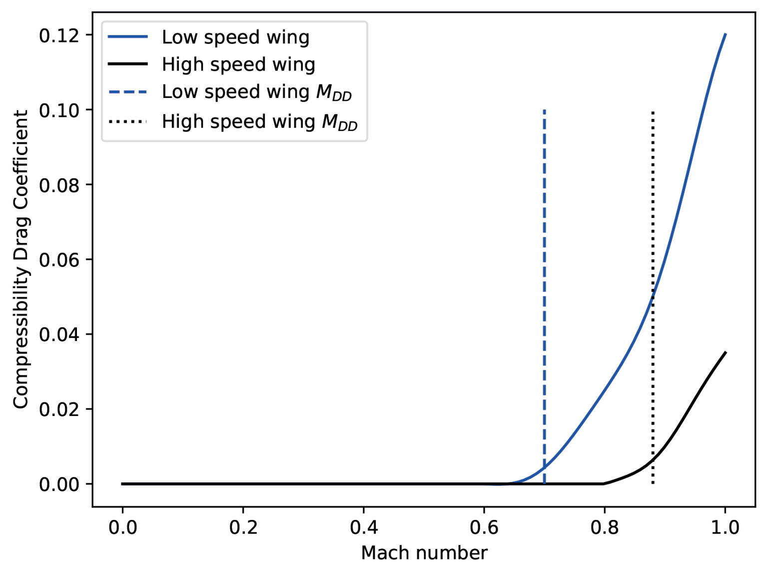
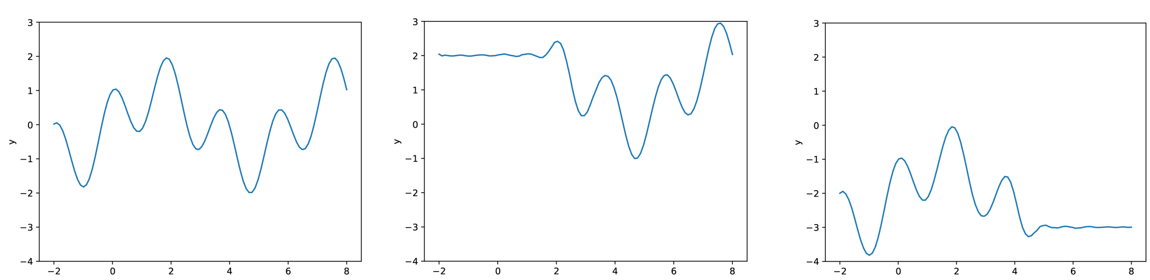


Figure 1: Sample curves showing the evolution of compressibility drag coefficient with the Mach number highlighting the drag-divergence Mach (M_{DD}) numbers between a low-speed and a high-speed wing (adapted from [23]).

which this happens is the drag-divergence Mach number M_{DD}. If we use source data from a high-speed wing model on a low-speed wing target problem, we will have large errors if the model is used across the whole design space. However, at Mach numbers below 0.6, models built on both the target and source data will produce equivalent wave drag results. Therefore, we are interested in methods to enable weighting of source models in ensemble of surrogates in specific areas in the design space. We present an illustrative numerical example to highlight our motivations in Figure 2.



(a) Target function. **(b) Source 1 function.** **(c) Source 2 function.**

Figure 2: Numerical example illustrating motivations for methods to divide a target problem's design space and to use posterior information from source surrogates.

We consider a target problem defined by a one-dimensional function $f(x) = \sin x + \cos \frac{10x}{3}$, with $x \in [-2, 8]$. We are interested in using two source functions: the first is defined as $f(x) + 1$ when

$x > 1.8$ and $2 + \varepsilon$ otherwise, and the second is defined as $f(x) - 2$ when $x < 4.5$ and $-2 + \varepsilon$ otherwise, ε is a noise term. Both of the source functions have a similar shape to the target function in only a region of the design space as shown in Figure 2 (b) and Figure 2 (c). We also note that the source functions do not share the same output values with the target function even in regions of the design space that their shapes are similar. Therefore, transfer learning methods must be able to use source surrogate models only in specific regions of the design space. We will use this example as a running example to demonstrate our methods in Section 3.

We summarize the motivating aircraft design scenarios as follows:

Scenario 1: an aircraft designer is interested in creating surrogate models of a computationally expensive discipline by transferring knowledge from other disciplines.

Scenario 2: an aircraft designer is interested in combining the prediction capabilities of multiple surrogate models that are valid in only specific regions of the design space.

To that end, we select the use of ensemble of surrogates when we have multiple surrogate models built using datasets from source problems and a single dataset from the target problem as depicted in Figure 3. This ensemble of surrogates must be able to perform any of the motivating scenarios described above. There exists several methods to apply transfer learning on surrogate modeling other than the use of ensembles of Gaussian processes. Authors in [33, 63] used a single surrogate model for the target problem which can be used and adapted based on prior information from existing Gaussian processes. In [63], the authors achieved this by applying different kernels: a squared exponential kernel for points in the target problem, and a nearest neighbor kernel for points in the source problems. Several methods also exist for transfer learning using multi-output Gaussian processes [2, 49, 61]. The authors in [28] used a so-called deep Gaussian process [12] by projecting the target data on a source Gaussian process which was then linearly combined with the second layer of a two-layer deep Gaussian process. Another popular use of transfer learning in surrogate modeling is the use of neural networks, a type of surrogate models that is comprised of multiple layers of interconnected nodes [51]. However, we consider these methods inadequate for the problems of interest in this research due to their higher computational costs compared to the ensemble of surrogates approach. In the following sections, we

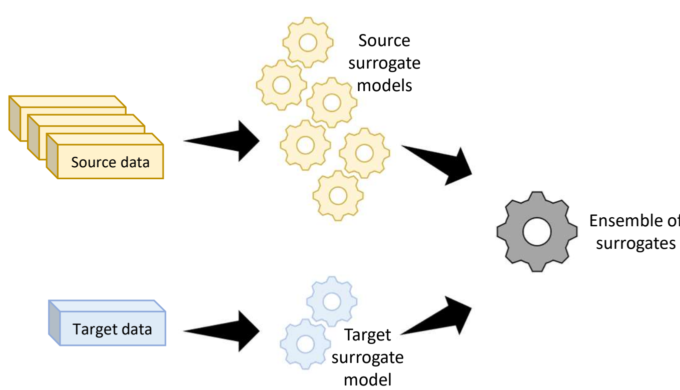


Figure 3: Illustration of transfer learning ensemble of surrogates.

present descriptions of the two aircraft design surrogate modeling problems based on Scenarios 1 & 2 that will be used in this paper as the application problems. We note that Scenario 3 is not applied in this paper and will be part of future work.

2.1 Aircraft noise prediction

Aircraft noise is a major consideration for aircraft and engine designers due to its impact on local communities and natural habitats around airports. Noise sources from aircraft are complex and interdependent as presented in Figure 4 (adapted from [65]). Predicting aircraft noise is a complex task and is computationally expensive for higher fidelity methods [18, 65]. Furthermore, validation tests of noise prediction methods are also expensive and rely on a variety of ground and flight tests. In this scenario, we assume that a small set of input-output data for the target problem, aircraft approach noise, is present; however, an aircraft acoustics model is not available to produce estimates for aircraft approach noise for new sets of inputs. Aircraft noise during the approach flight phase, i.e., aircraft approach noise, is always higher than other reference flight phases, i.e, lateral and flyover noise, and is typically measured using the EPNL, effective perceived noise level, metric [18]. We hypothesize that aircraft approach noise is correlated with aircraft maximum takeoff weight (MTOW) as a heavier aircraft requires a larger airframe, engines, high lift devices, and landing gear. In this problem, we aim to use transfer learning to build a surrogate model of aircraft approach noise based on a set of input variables and a surrogate model of MTOW.

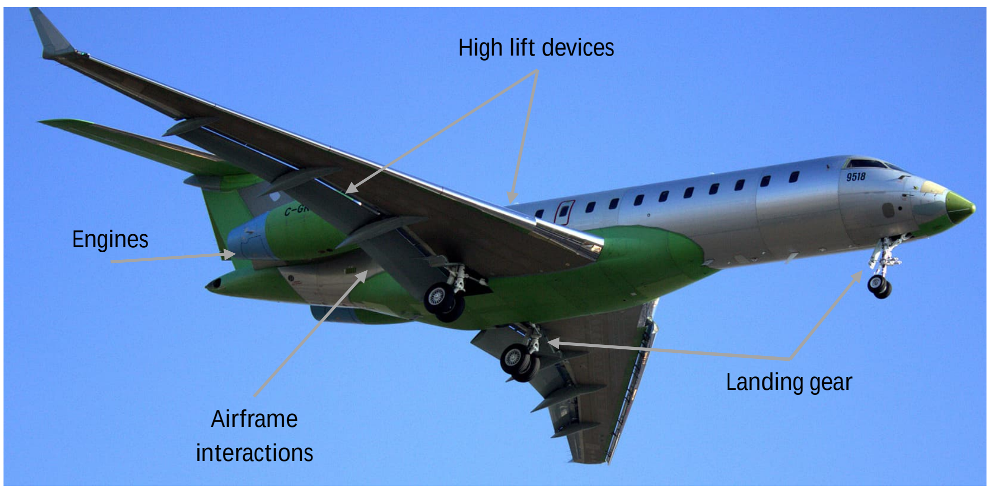


Figure 4: Aircraft approach noise sources (adapted from [65]). Photo by Yan Gouger.

2.2 Slats configuration impact on aircraft weight

For the second aircraft design problem we are interested in estimating the MTOW of an aircraft depending on whether the aircraft has slats, i.e., slatted, or not, i.e., unslatted. Aircraft wing slats are a type of leading edge high lift devices, an example of the Airbus A321 aircraft slat is shown in Figure 5, that allow an aircraft's wing to operate at a higher angle of attack without stalling therefore allowing the aircraft to takeoff and land on shorter runways and at lower speeds [45]. By enabling an aircraft to fly at lower speeds at takeoff and landing than unslatted aircraft, slats allow designers to reduce the size of an aircraft's wing which significantly reduces its cruise drag especially for high-speed transonic aircraft. Slats are typically attached to the wing's front spar (F/S) by means of actuators and tracks where multiple competing constraints drive their design and implementation. Although slats significantly improve the low speed, ground performance, and high-speed cruise drag of an aircraft, their integration comes with significant challenges such as actuation installation, anti-icing, cost and reliability. Therefore, the decision for installing slats or not on an aircraft is complex and multi-disciplinary which necessitate the use of optimization and aircraft level trade studies. In this problem, we are interested in performing analyses that predict the MTOW of an aircraft based on aircraft sizing input variables in addition to whether an aircraft is slatted or unslatted. Consider a scenario where we have two sets of simulation models: M1) the first model correctly models a slatted aircraft and predicts its MTOW but produces incorrect results if the aircraft is unslatted, and M2) the second model behaves conversly by predicting correct MTOW only for unslatted aircraft. We are interested in running an optimization where slat configuration is a design variable and we combine the computationally expensive simulation models of M1 and M2 into the same framework. Our goal is to perform transfer learning in the new framework using existing results we had obtained using either M1 or M2 solely. The transfer learning methodology must be able to capture the correct behavior of M1 and M2 in their respective applicable design spaces.

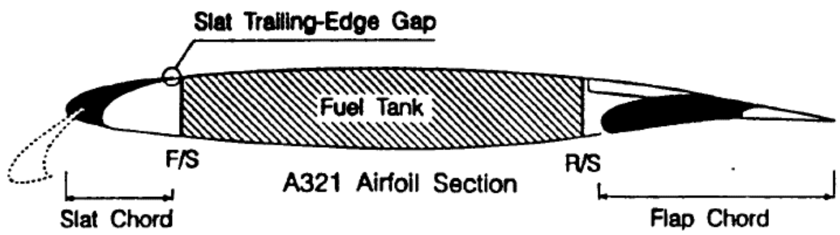


Figure 5: Airbus A321 wing cross section showing a deployed slat [45].

3 Methodology

As described in Section 2, we focus in this work on ensembles of Gaussian processes. In the following sections, we describe the methodologies used to develop an ensemble of Gaussian processes using transfer learning. Based on the motivations described above, we identify the following that our methodology must achieve:

1. Build Gaussian processes from the source data but adjusted to the output of a target problem.
2. Divide the design space of a target problem based on the performance of source surrogate models.
3. Determine appropriate weighting of source models for both surrogate modeling.

To achieve the above, we first propose an approach to transfer the prior mean function from a source surrogate model to a target model in Section 3.1. Secondly, we present a method to divide the design space preparing for a mixture of experts algorithm in Section 3.2. Lastly, we propose a multi-criteria weighting methodology to enable expert selection in each region of the design space in Section 3.3. In the literature, there are several methods for the weighting of the ensemble surrogate model [10, 17, 22, 60, 62, 64]. However, these methods are not flexible for use in transfer learning for surrogate modeling and optimization. In this work, we develop a multi-criteria approach that can be altered depending on the use case at hand. The selectable criteria are based on the shape, accuracy, and variance of the surrogates in an ensemble. We use Gaussian processes in all of the surrogates in this work to obtain variance information to enable ensemble weighting and to use them in Bayesian optimization. Finally, we combine the steps above into an algorithm for building an ensemble of surrogates using transfer learning. The notation we use in this work is defined in Table 1.

Table 1: Notations in transfer learning for surrogate modeling.

Symbol	Definition
$\mathrm{DoE}_{\mathrm{S}}^{1,\dots,N}$	Design of experiments datasets containing input/output pairings from source problems $1,...,N$
$\mathrm{DoE}_{\mathrm{T}}$	Design of experiments dataset containing input/output pairings from the target problem
X_{T}	Input vector data from $\mathrm{DoE}_{\mathrm{T}}$
Y_{T}	Output vector data from $\mathrm{DoE}_{\mathrm{T}}$
$\hat{y}_{\mathrm{S}}^{1,\dots,N}$	$1,...,N$ source surrogate models built using source datasets $\mathrm{DoE}_{\mathrm{S}}^{1,\dots,N}$
$\hat{y}_{\mathrm{S}'}^{1,\dots,N}$	$1,...,N$ modified source surrogate models, $\hat{y}_{\mathrm{S}}^{1,\dots,N}$, using target dataset $\mathrm{DoE}_{\mathrm{T}}$
$\hat{y}_{\mathrm{t}}$	Target surrogate model built using target dataset $\mathrm{DoE}_{\mathrm{T}}$

3.1 Transfer of prior

When transferring a source model to a target problem, the use of a source model *as-is* or its prior, therefore maintaining the shape of the source model, is well established in literature We define the shape of a model as the behaviour of the model's outputs corresponding to changes in inputs values. In this work, we also are interested in the value predictions of source models and not just their shapes. Therefore, we propose to transfer source model posteriors, $\hat{y}_{\mathrm{S}}$, and adjust their predictions to target data $\mathrm{DoE}_{\mathrm{T}}$ similar to the approach proposed in [52]. We then use these posteriors as the prior mean functions m of a target problem, such that $m(x) = \alpha\hat{y}_{\mathrm{S}}(x)$, where $\hat{y}_{\mathrm{S}}$ is the posterior of a source model and α is a scaling term. Starting from the squared exponential covariance function, the authors in [52] proposed a new covariance function as follows

$$k^{'}(x_i, x_j) = \sigma_0^2 \exp\left[-\frac{||x_i - x_j||^2}{2l^2}\right] \exp\left[-\frac{(\hat{y}_{\mathrm{S}}(x_i) - \hat{y}_{\mathrm{S}}(x_j))^2}{2l^2_{\hat{y}_{\mathrm{S}}}}\right], \tag{1}$$

where hyperparameters σ_0 and l are the variance and characteristic length-scale of the covariance respectively, and $l_{\hat{y}_{\mathrm{S}}}$ is the length-scale obtained from $\hat{y}_{\mathrm{S}}$ and are calculated using maximum likelihood estimation [1]. The covariance function in Equation (1) may potentially change the shape of the source models $\hat{y}_{\mathrm{S}}$ between two points x_i and x_j since the covariance function $k^{'}$ and hyperparameters σ_0 and l are determined based on $\hat{y}_{\mathrm{S}}(x_i)$ and $\hat{y}_{\mathrm{S}}(x_j)$. However, in some cases, designers are interested in maintaining the shape of the source model without modifying the covariance function. In addition, the work in [52] developed their approach specifically for use in a Bayesian optimization context which does not need to adjust the bias of the source models since the Gaussian process hyperparameters are fit to the new observations. To that end, we propose an approach for adjusting a source model $\hat{y}^n_{\mathrm{S}}$ considering both a scaling factor and a bias.

$$\hat{y}^n_{\mathrm{S}'}(x) = \alpha_n \hat{y}^n_{\mathrm{S}}(x) + \beta_n, \tag{2}$$

where the scaling term α_n and the bias term β_n are constant parameters that can be fitted to the target data $\mathrm{DoE}_{\mathrm{T}}$ to create a modified set of source models as described in Algorithm 1. Alternatively, one could still use the covariance function from Equation (1) with a prior mean function from Equation (2). The output of Algorithm 1 is N modified source surrogate models that require aggregation into an ensemble, which is addressed in the following sections.

Algorithm 1: Source surrogate model with posterior information.

Input: Source datasets $\mathrm{DoE}^{1,\ldots,N}_{\mathrm{S}}$ and target dataset $\mathrm{DoE}_{\mathrm{T}} : (\mathrm{X}_{\mathrm{T}}, \mathrm{Y}_{\mathrm{T}})$
Output: Modified source models $\hat{y}^{1,\ldots,N}_{\mathrm{S}'}$.

```
for n = 1, ..., N do
    Build source model ŷ^n_S based on DoE^n_S
    Use the posterior of the ŷ^n_S to calculate ŷ^n_S(X_T).;
    Calculate (α_n, β_n) = argmin_{α,β∈ℝ} ε(α, β) where ε is an error metric between ŷ^n_S'(X_T) and Y_T, ;
    Calculate the hyperparameters σ^n_0 and l^n.;
    Set modified source surrogate model per Equation (2) and Equation (1);
end
```

3.2 Mixture of experts approach

To address the problem of fixed weights across the whole design space for ensembles of surrogates, we propose a mixture of experts approach that divides the design space into clusters and calculates the probability for each source model being the correct model for each cluster. Mixture of experts is based on the divide-and-conquer principle in which the problem space is divided between a number of weighted experts, i.e., surrogate models, that are supervised by a gating network. The authors in [27] presented a general architecture of mixture of experts for supervised learning. This architecture is a tree structure where the tree splits are called a gating network. Each gating network produces soft tree splits until the terminal leaves which produce an output by means of a generalized linear model. The parameters of the different generalized linear models are estimated using the Expectation-Maximization (EM) algorithm [13, 37] on the input-output space. In this work, we use source models that are already developed, and we only need to develop a gating network without redeveloping the existing surrogate models. In [4], the authors developed a mixture of experts algorithm aiming to increase the accuracy of a function approximation by replacing a single global model by a weighted sum of local experts. The approach is based on a partition of the problem domain into several subdomains via clustering algorithms followed by a local expert training on each subdomain. Clustering does not use generalized linear models but Gaussian mixture models obtained using the EM algorithm which allows for a smooth transition between clusters. They use a *latent* discrete random variable κ that indicates which component of the mixture is to be used. The authors in [31] applied the mixture of experts algorithm from [4] on aircraft aerodynamic prediction problems. The classical expression of a mixture of experts model is presented as follows

$$\hat{y}(x) = \sum_{i=1}^{K} \mathbb{P}(\kappa = i | X = x)\hat{y}_i(x), \tag{3}$$

where K is the number of clusters, $\kappa \in \{1, ...K\}$ denotes the discrete random variable associated with the clusters, $\mathbb{P}(\kappa = i|X = x)$ is the probability of a point x to lie in cluster with index i, and $\hat{y}_i$ is the local expert built on cluster i knowing that a Gaussian $X = x$. The ideal number of clusters is problem dependent and can be obtained using the EM algorithm by utilizing the so-called Bayesian Information Criterion or Akaike Information Criterion [7]. The gating network probability is derived in [4]. Assuming a Gaussian mixture GM of K components, the EM algorithm [13, 37] is used to estimate the parameters of each Gaussian $k \in 1...K$ based on (X, Y) pairings:

$$GM : (X, Y) \sim \sum_{k=1}^{K} \alpha_k \mathcal{N}(\mu_k, \Gamma_k), \tag{4}$$

where α_k is the mixture parameter for each Gaussian such that $\alpha_k \in [0, 1]$ and $\sum_{k=1}^{K} \alpha_k = 1$, μ_k is the mean of the Gaussian distribution k of a Gaussian $\mathcal{N}(\mu_k, \Gamma_k)$, and Γ_k is its variance-covariance matrix. Using the EM algorithm, the law of κ knowing that $X = x$ and without knowing its associated output Y is obtained as follows

$$X \sim \sum_{k=1}^{K} \alpha_k \mathcal{N}(\mu_k^X, \Gamma_k^X), \tag{5}$$

where Γ_k^X and μ_k^X are the variance-covariance matrix and mean of X for Gaussian k, respectively.

In this work, we propose to cluster the design space based on combinations of inputs values x and an error metric $\mathcal{E}$ of each source model on the target data $\mathrm{DoE_T}$. Therefore, the clustering matrix M of dimensions $p \times (n + N)$ is defined as

$$M = \begin{bmatrix} x_{11} & \cdots & x_{n1} & \epsilon_{11} & \cdots & \epsilon_{N1} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ x_{1p} & \cdots & x_{np} & \epsilon_{1p} & \cdots & \epsilon_{Np} \end{bmatrix}, \tag{6}$$

where n is the dimension of the input vector x, N is the number of source models, p is the number of points in $\mathrm{DoE_T}$, x_{np} is the value of the n^{th} dimension of a point with index p of $\mathrm{DoE_T}$, and ϵ_{Np} is the value of the N^{th} source model error at a point with index p of $\mathrm{DoE_T}$. In this work, we use an average absolute error for ϵ; however, other metrics can be used depending on the use case needs. Consequently, the Gaussian mixture (X, Y) used in the EM algorithm is set to $(X, \mathcal{E})$ which is built based on the input vector x and error vector ϵ as per the clustering matrix M. Once the EM algorithm is used to calculate the Gaussian mixture parameters, the probability for a given point $z = (x, \epsilon)$ to lie within a cluster i adapted from the derivation in [4] as a so-called *hard* clustering method as follows.

$$\mathbb{P}(\kappa = i | (X, \mathcal{E}) = (x, \epsilon)) = \frac{\det(\Gamma_i)^{-1/2}\alpha_i \exp[-\frac{1}{2}(z-\mu_i)^T\Gamma_i^{-1}(z-\mu_i)]}{\sum_{k=1}^{K}\det(\Gamma_k)^{-1/2}\alpha_k \exp[-\frac{1}{2}(z-\mu_k)^T\Gamma_k^{-1}(z-\mu_k)]}. \tag{7}$$

Each cluster $i \in [1, ..., K]$ is then defined where each point $(x,\ \epsilon)$ yields the highest probability from Equation (7) as follows,

$$i = \text{argmax}_{j\in[1,...,K]}\ \mathbb{P}(\kappa = j | (X, \mathcal{E}) = (x, \epsilon)). \tag{8}$$

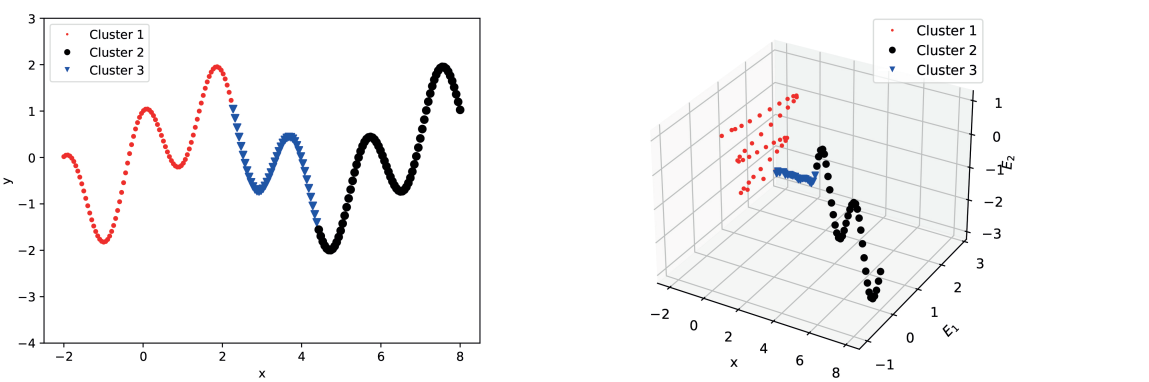
The probability of a point being in a cluster i is then calculated using only the input vector x using a so-called *smooth clustering* setting as follows,

$$\mathbb{P}(\kappa = i | X = x) = \frac{\det(\Gamma_i^X)^{-\frac{1}{2}}\alpha_i \exp[-\frac{1}{2}(x-\mu_i)^T\Gamma_i^{X-1}(x-\mu_i)]}{\sum_{k=1}^{K}\det(\Gamma_k^X)^{-\frac{1}{2}}\alpha_k \exp[-\frac{1}{2}(x-\mu_k^x)^T\Gamma_k^{X-1}(x-\mu_k^X)]}. \tag{9}$$

Alternatively, in a *hard* clustering setting, $\mathbb{P}(\kappa = i|X = x)$ in Equation (3) is replaced by $\mathbb{1}_{\mathbb{P}_i^X}(x)$ given by

$$\mathbb{1}_{\mathbb{P}_i^X}(x) = \begin{cases} 1.0, & \text{if } i = \text{argmax}_{j=1,...,K}\mathbb{P}(\kappa = j|X = x), \\ 0.0, & \text{otherwise.} \end{cases} \tag{10}$$

We validate this approach on the one-dimensional function from Figure 2. We build two source surrogate models based on the source functions in Figure 2 (b) and Figure 2 (c) and apply Algorithm 1 from Section 3.1 to obtain modified source surrogates, $\hat{y}^1_{S'}$ and $\hat{y}^2_{S'}$. We then calculate the errors, $\mathcal{E}_1$ and $\mathcal{E}_2$ of $\hat{y}^1_{S'}$ and $\hat{y}^2_{S'}$ respectively, when compared to the target function $f(x)$ across the whole design space using 100 equally spaced points $(X, \mathcal{E})$. We create the clustering matrix M in Equation (6) using X and $\mathcal{E}$. Finally, we set the number of clusters to three, apply the EM algorithm, calculate the *hard* clustering probabilities from Equation (10), and obtain the clusters from Equation (8). The results of the clusters and clustering matrix are presented in Figure 6 demonstrating the ability of the proposed method in correctly dividing the design space based on the accuracy of the source surrogate models.



(a) Clusters visualization on the target function. **(b) Visualization of the clustering matrix M comprised of X, $\mathcal{E}_1$ and $\mathcal{E}_2$.**

Figure 6: Illustration of clustering on a one-dimensional target function with 3 clusters and 2 source surrogate models.

The resulting mixture of experts algorithm for clustering based on input vectors and source model errors is presented in Algorithm 2. We note that the Gaussian mixture clustering method used in this work may not be accurate for non-Gaussian functions. In application cases where such functions exist, future work may adapt different clustering approaches [50, 66] to obtain the clusters and the gating network.

Algorithm 2: Design space clustering and mixture of experts gating network.

Input: Modified source models $\hat{y}_{\mathrm{S}'}^{1,\dots,N}$, target dataset $\mathrm{DoE_T} : (\mathrm{X_T}, \mathrm{Y_T})$, and number of clusters K
Output: Clusters K and gating network definition

1. Calculate the errors $\mathcal{E}_n$ between each modified source model $\hat{y}_{\mathrm{S}'}^{n}(\mathrm{X_T})$ and $\mathrm{Y_T}$ for $n = 1, ..., N$. ;
2. Assemble the clustering matrix M in Equation (6) using $\mathrm{X_T}$ and the errors $\mathcal{E}_{1,\dots,N}$.
3. Apply EM algorithm to M with clusters K to get estimates of the Gaussian mixture parameters α_k, μ_k and Γ_k for $k = 1, ..., K$.
4. Perform the hard clustering step of the data using Equation (7) and Equation (8) to obtain the set of clusters K.
5. Set the gating network $\mathbb{P}(\kappa = i|X = x)$ at any point x for each cluster $i \in K$ using Equation (9)

Although the separation of clustering and learning in [4] is beneficial for our transfer learning problem, this algorithm builds a set of different surrogate model types on the same data and calculates a weight for each surrogate based on its performance in each subset of the design space relative to a predefined metric. This approach does not match the needs of transfer learning problems where the intended local experts are already predefined. In addition, mixture of experts algorithms in [4, 31] assume that the number of surrogate models is pre-set based on the selected types of surrogates, and that the cluster definition is a prior step to build the surrogate models.

In the transfer learning problem setting, the number of source models may not match the number of clusters and the clusters are dependent on the source models. Therefore, we propose an algorithm that first divides the problem space based on $(X, \mathcal{E})$ to obtain the gating network. Then, we add a term to obtain the probability that each source model is the correct model to be used in each cluster. The expression defining the prediction $\hat{y}$ at a point x of our proposed algorithm is given by

$$\hat{y}(x) = \sum_{i=1}^{K}\sum_{j=1}^{N} \mathbb{P}(\kappa = i|X = x)\mathbb{P}(\hat{y}_j = \hat{y}_i)\hat{y}_j(x), \tag{11}$$

where K is the number of clusters, N is the number of source models, i is the index of existing clusters, $\hat{y}_i$ is the Gaussian model representing the indexed cluster, and $\hat{y}_j$ is the j^{th} source model. Equation (11) can be considered a *smooth* prediction setting assuming both $\mathbb{P}(\kappa = i|X = x)$ and $\mathbb{P}(\hat{y}_j = \hat{y}_i)$ are smooth. Alternatively, a *hard* prediction setting can re-written as follows

$$\hat{y}_{\text{hard}}(x) = \sum_{i=1}^{K}\sum_{j=1}^{N} \mathbb{1}_{\mathbb{P}_i^X}(x)\mathbb{1}_{\mathbb{P}_i^j}(j)\hat{y}_j(x), \tag{12}$$

where $\mathbb{1}_{\mathbb{P}_i^j}$ is determined as the source model with highest probability to represent the model of cluster with index i, and $\mathbb{1}_{\mathbb{P}_i^X}$ is the *hard* clustering method from Equation (10).

$$\mathbb{1}_{\mathbb{P}_i^j}(j) = \begin{cases} 1.0, & \text{if } i = \text{argmax}_{j=1,\dots,K}\mathbb{P}(\hat{y}_j = \hat{y}_i), \\ 0.0, & \text{if } i \neq \text{argmax}_{j=1,\dots,K}\mathbb{P}(\hat{y}_j = \hat{y}_i). \end{cases} \tag{13}$$

To obtain the standard deviation $\hat{s}$ of the ensemble of surrogates $\hat{y}$, we present two methods:

1. target surrogate model variance $\hat{s}_t$ as proposed in [60]

$$\hat{s}^2(x) = \hat{s}_t^2(x). \tag{14}$$

2. weighted ensemble of surrogates variances based on *smooth* clustering and prediction as follows

$$\hat{s}^2(x) = \sum_{i=1}^{K}\sum_{j=1}^{N} (\mathbb{P}(\kappa = i|X = x))^2(\mathbb{P}(\hat{y}_j = \hat{y}_i))^2\hat{s}_j^2(x), \tag{15}$$

where $\hat{s}_j(x)$ is the standard deviation of the j^{th} source model.

In the following section, we complement the methods presented so far to obtain $\mathbb{P}(\hat{y}_j = \hat{y}_i)$, the probability that source model j is the correct model to be used in cluster i.

3.3 Multi-criteria weighted ensembles

In existing literature, the use of transfer learning in ensembles of Gaussian processes relies on the mis-ranked or discordant pairs approach [16, 30, 60]. The idea is to select all paired combinations of a source model's predictions and the target data and estimate how often this prediction and the target point agree on the ranking of outputs. This approach promises to measure a source model's ability to generalize to the target data based on its shape. Such methods are appropriate for Bayesian optimization (for the objective function only) where the shape of a function is more important than the actual prediction values. However, the use of transfer learning for applications other than objective functions in optimization requires the use of different criteria for model selection. We propose a multi-criteria ranking approach that can be selected depending on the intended application. Even for the case of Bayesian optimization, the ideal criterion may not always depend on the shape of a function. For example, handling constraints requires model selection based on accuracy and not just the shape of the constraint functions. The mis-ranked pairs approach for model selection is more useful for performing acquisition function optimization since we are interested in the shape of the predicted function rather than its values. In [16], the authors developed an ensemble of Gaussian processes transfer learning method where source models, along with the target model, are weighted based on the shape of the function, i.e., mis-ranked pairs. They used a loss function to calculate the weights of each source model when applied to the target data.

$$\mathcal{L}(\hat{y}_{\mathrm{S}}, \mathrm{DoE}_{\mathrm{T}}) = \sum_{m=1}^{n} \sum_{k=2}^{n} \mathbb{1}((\hat{y}_{\mathrm{S}}(x_m^t) < \hat{y}_{\mathrm{S}}(x_k^t)) \oplus (y_m^t < y_k^t)), \tag{16}$$

where $\oplus$ is the exclusive-or operator, $\mathrm{DoE}_{\mathrm{T}}$ is the input-output data $(\mathrm{X}_{\mathrm{T}}, \mathrm{Y}_{\mathrm{T}})$ of the target problem T with n points, and $\hat{y}_{\mathrm{S}}$ is a source model. The posterior of each source model is then used at the target data creating a set of samples where the ranking losses are evaluated using Equation (16). Weights for each source model are then calculated by the summation of the samples' ranking losses divided by the number of samples. In [60], the authors developed a similar ensemble of Gaussian processes algorithm also based on function shape. However, to calculate the ensemble weights of the source models, they first calculate the ratio τ of discordant pairs to total pairs n of $\mathrm{DoE}_{\mathrm{T}}$ [29] between source models and the target model. Then they use a weighted average to predict the combined output of the source and target models. The Epanechnikov quadratic kernel [15] is used as follows

$$\kappa_{\rho}(\chi_S, \mathrm{DoE}_{\mathrm{T}}) = \delta(\frac{\tau(\chi_S, \mathrm{DoE}_{\mathrm{T}})}{\rho}) \tag{17}$$

with

$$\delta(t) = \begin{cases} \frac{3}{4}(1-t^2) & \text{if } t \leqslant 1 \\ 0 \quad \text{otherwise} \end{cases} \tag{18}$$

and

$$\tau(\chi_S, \mathrm{DoE}_{\mathrm{T}}) = \frac{\sum_{m=1}^{n} \sum_{k=2}^{n} \mathbb{1}((y_m^s < y_k^s) \oplus (y_m^t < y_k^t))}{n}, \tag{19}$$

where $\rho > 0$ is a predefined bandwidth, $\mathrm{DoE}_{\mathrm{T}}$ is the input-output data of the target problem T, and χ_S are the data predictions Y^s on the inputs X_{T} of $\mathrm{DoE}_{\mathrm{T}}$ using a source model S with n points.

In this work, we propose a multi-criteria approach to obtain the probability of a surrogate model j being a correct representation in a cluster i. A summation of weighted multi-criteria $C(\hat{y}_j = \hat{y}_i)$ to calculate a score for a model $\hat{y}_j$ being a correct model in cluster i is presented as

$$C(\hat{y}_j = \hat{y}_i) = \sum_{l=1}^{NC} w_l c_l(\hat{y}_j = \hat{y}_i), \tag{20}$$

where $w_l \in [0, 1]$ for $l = 1, ..., NC$ are the weights assigned to the selected criteria such that $\sum_{l=1}^{NC} w_l = 1$, c_l is the measure of the selected criterion, NC is the number of measured criteria. We propose the following set of criteria to select from: shape, accuracy, and variance.

The measure of the selected criteria is up to the user and depends on the type of algorithm where the surrogate model is used. The shape criterion denotes a measure of the degree to which a source surrogate model is representative relative to the shape of the target data. The methods from [16, 30, 60] are all considered different types of shape criteria. In this work, we use the discordant pairs method from Equation (17) [60] as the shape criterion.

The accuracy and variance criteria compare the values of the model predictions with the values of the target data using modified versions of the shape criterion by replacing τ in Equation (17) by τ_a and τ_v respectively. Similarly, bandwidths for the accuracy and the variance criteria, ρ_a and ρ_v respectively, are assigned instead of ρ in Equation (17).

For the accuracy criterion, we use a measure of the absolute relative error metric across the target dataset $\mathrm{DoE}_{\mathrm{T}}$ to obtain τ_a as follows

$$\tau_a(\chi_S, \mathrm{DoE}_{\mathrm{T}}) = \frac{1}{n} \sum_{m=1}^{n} \mathbb{1} \left(\frac{|y_m^s - y_m^t|}{|y_m^t|} > \epsilon_{max} \right), \tag{21}$$

where ϵ_{max} is a predefined maximum relative error.

Similarly for the variance criterion, we propose the use of the same kernel in Equation (17) with $\tau = \tau_v$ and $\rho = \rho_v$ as follows

$$\tau_v(\chi_S, \mathrm{DoE}_{\mathrm{T}}) = \frac{1}{n} \sum_{m=1}^{n} \mathbb{1} \left(\frac{\sigma_m^s}{y_{max}^t} > \sigma_{max} \right), \tag{22}$$

where σ_{max} is a predefined maximum relative variance, y_{max}^t is the maximum observed absolute output value in $\mathrm{DoE}_{\mathrm{T}}$, and σ_m^s is the variance of point m measured using source model S.

Once the criteria measures are calculated, we assume that the probability of a model being the correct model over a cluster is proportional to its criteria measure in this cluster. Therefore, the probability of a source model j predicting a correct output of target data in cluster i is then estimated as

$$\mathbb{P}(\hat{y}_j = \hat{y}_i) \approx \frac{C(\hat{y}_j = \hat{y}_i)}{\sum\limits_{n=1}^{N} C(\hat{y}_n = \hat{y}_i)}, \tag{23}$$

where N is the number of source models.

So far, we have presented the prediction of the mean and variance of the ensemble of surrogates while ignoring the predictions of the target model $\hat{y}_{\mathrm{t}}$ assuming that its predictions are inaccurate due to insufficient data. Alternatively, one could assign a fixed probability of the target model $\hat{y}_{\mathrm{t}}$ and include it in the predictions of the ensemble of surrogates $\hat{y}$. The corresponding second probability term is defined as

$$\mathbb{P}(\hat{y}_m = \hat{y}_i) \approx \frac{C(\hat{y}_m = \hat{y}_i)}{\sum\limits_{n=1}^{N} C(\hat{y}_n = \hat{y}_i) + C(\hat{y}_{\mathrm{t}} = \hat{y}_i)}, \tag{24}$$

where $\hat{y}_m$ is the m^{th} model in a list comprised of N source models and 1 target model, and $C(\hat{y}_{\mathrm{t}} = \hat{y}_i)$ is the fixed score for the target model. In the case of the Epanechnikov quadratic kernel, $C(\hat{y}_{\mathrm{t}} = \hat{y}_i)$ is set to 0.75 in [60]. Equation (23) or (24) can then be used in the second probability term of Equation (11) from Section 3.2 and complete the prediction of the mixture of experts.

3.4 Ensemble of surrogates using transfer learning

The ensemble of surrogates using transfer learning algorithm can be summarized based on the methodologies described above. The algorithm starts by building a set of source surrogate models using the available source data. The algorithm modifies the source models per the approach proposed in Section 3.1, then develops a mixture of experts as described in Section 3.2. Finally, the weighting of each surrogate in the ensemble is determined using the multi-criteria approach we proposed in Section 3.3. The detailed procedure to obtain the ensemble of surrogates is described in Algorithm 3.

Algorithm 3: Ensembles of surrogate models using transfer learning.

Input: Source datasets $\mathrm{DoE}_{\mathrm{S}}^{1,\dots,N}$, target dataset $\mathrm{DoE}_{\mathrm{T}}$, number of clusters K, selected criteria NC, and criteria weights w_l and bandwidths ρ_l for $l \in 1, \dots NC$
Output: Ensemble of surrogates using transfer learning

```
1  Build target surrogate model ŷ_t using DoE_T.;
2  for n = 1, ..., N do
3      Build n^th source surrogate model ŷ^n_S using DoE^n_S.;
4      Use the posterior of ŷ^n_S and DoE_T to create a modified source surrogate model ŷ^n_S' using Algorithm 1.;
5  end
6  Create a set of K clusters using the mixture of experts approach defined in Algorithm 2 to obtain the gating
     network P(κ = i|X = x).;
7  for k = 1, ..., K do
8      for n = 1, ..., N do
9          Calculate the criteria scores C(ŷ^n_S' = ŷ_k) of each ŷ^n_S' in each cluster k using Equation (20).;
10         Calculate the probability P(ŷ^n_S' = ŷ_k) of each ŷ^n_S' in each cluster k using Equation (23)
             or Equation (24).;
11     end
12 end
13 Return the ensemble of surrogates model (ŷ, ŝ).
```

After creating the ensemble of surrogates, clusters, and model probabilities for each cluster of the mixture of experts, a prediction $\hat{y}(x)$ using input data x is calculated using Equation (11) or Equation (13), and standard deviation $s(x)$ is calculated using Equation (14) or Equation (15). We note that users can elect to maintain source surrogate models without modification by skipping line 4 in Algorithm 3 and assigning $\hat{y}^n_{\mathrm{S}'} = \hat{y}^n_{\mathrm{S}}$ for the rest of the algorithm.

We presented in Equation (20) the weights, w_l for $l = 1, \dots, NC$ associated with each criterion c_l. These weights can be obtained by formulating an optimization problem that minimizes a prediction error metric of choice using the ensemble of surrogates model on target data. In [3], the authors proposed a similar metric named *Penalized Predictive Score* (PPS), and they use PPS directly as an optimization objective to obtain the weights within (we note that PPS uses different criteria than those we propose in Section 3.3). In our work, we need to be able to calculate common weights for all source models and all clusters to build the ensemble of surrogates. Therefore, simply minimizing a predictive error metric on a single surrogate model as in [3] does not work. Instead, we propose an optimization problem that encompasses all of the transfer learning problem including all clusters K and source models N to obtain the weights w_l for $l = 1, \dots, NC$ associated with each criterion as follows,

$$\begin{cases} \underset{w \in [0,1]^{NC}}{\text{minimize}} & \displaystyle\sum_{i=1}^{K} \mathbb{E}(\hat{\mathrm{Y}}^i, \mathrm{Y}^i_{\mathrm{T}}, w) \\ \text{subject to} & \displaystyle\sum_{l=1}^{NC} w_l = 1, \end{cases} \tag{25}$$

where w is a vector of all w_l weights, and $\mathbb{E}(\hat{\mathrm{Y}}^i_n, \mathrm{Y}^i_{\mathrm{T}}, w)$ is an error metric between the output of the ensemble of surrogates model, $\hat{\mathrm{Y}}^i$, and the target data, $\mathrm{Y}^i_{\mathrm{T}}$, that lie in cluster i.

Alternatively, a designer may decide to set fixed weights or a variable weight selection strategy based on the envisioned transfer learning problem. For example, an unconstrained optimization problem benefits from the shape of a source surrogate model whereas a single point sizing problem benefits from a source surrogate model's accuracy and variance. Therefore, designers interested in optimization problems can assign higher weights for the shape criterion, whereas designers requiring accurate predictions from a surrogate can assign higher weights for the accuracy and variance criteria.

We also note that if a similar kernel to Equation (20) is used for criteria calculations, the bandwidths ρ_l for $l = 1, ..., NC$ associated with each criterion c_l can also be added as design variables in the formulation of Equation (25). The optimization problem can then be formulated as

$$\begin{cases} \underset{w,\rho}{\text{minimize}} & \displaystyle\sum_{i=1}^{K} \mathbb{E}(\hat{\mathrm{Y}}^i, \mathrm{Y}_{\mathrm{T}}^i, w, \rho) \\ \text{subject to} & \displaystyle\sum_{l=1}^{NC} w_l = 1, \end{cases} \tag{26}$$

where w is a vector of all w_l weights such that $w_l \in [0, 1]$ and ρ is a vector of all ρ_l bandwidths such that $\rho_l > 0$ for $l = 1, ..., NC$.

We note that the transfer learning algorithms presented in this work could be used by relying on source data that does not correlate with the target problem. In certain cases where source and/or data are not sufficient to create the ensembles of surrogates, inaccurate predictions may be expected. Therefore, it is important for users to conduct thorough design of experiments and assessments of the suitability of the source and target data prior to deployment of the transfer learning methods.

4 Numerical results

We present analytical and aircraft design examples to demonstrate the proposed methodologies in this work. We first explain the implementation details we used then present the illustration and aircraft design examples that verify each novel functionality.

4.1 Implementation details

The code is developed in Python 3 building on the following toolboxes: Scikit-learn v1.5.2 [41], SMT 2.0: Surrogate Modeling Toolbox [47], SMAC3 [32], RGPE [17], TST-R [60], and the Bayesian optimization library in [20]. All results are obtained using an Intel® Xeon® CPU E5-1650 v3 @ 3.50 GHz core and 32 GB of memory. The illustration tests the behavior of the following methods:

- Target surrogate model ($\hat{y}_{\mathbf{t}}$),
- Source surrogate model n ($\hat{y}^n_{\mathbf{S}}$),
- Modified source surrogate model n ($\hat{y}^n_{\mathbf{S}'}$).
- Ensemble of surrogates without clustering (Ensemble-TL),
- Ensemble of surrogates with *smooth* clustering and prediction from Equation (11) (Ensemble Smooth-C),
- Ensemble of surrogates with *hard* clustering and prediction from Equation (12) (Ensemble Hard-C).

4.2 Analytical examples

In this section, we present two analytical examples along with sensitivity analyses. In the first example, we illustrate the prior transfer algorithm with an ensemble of Gaussian processes (GP), i.e., Algorithm 3 without clustering in step 6. In the second example, we illustrate the clustering algorithm with an ensemble of GP, i.e., Algorithm 3 without source model modification in step 4.

4.2.1 Modified Bohachevsky functions

We use modified versions of the two-dimensional Bohachevsky functions [5, 26] such that the functions share similar shapes. We use three source functions, f_1, f_2, f_3, and one target function, f_t, as follows:

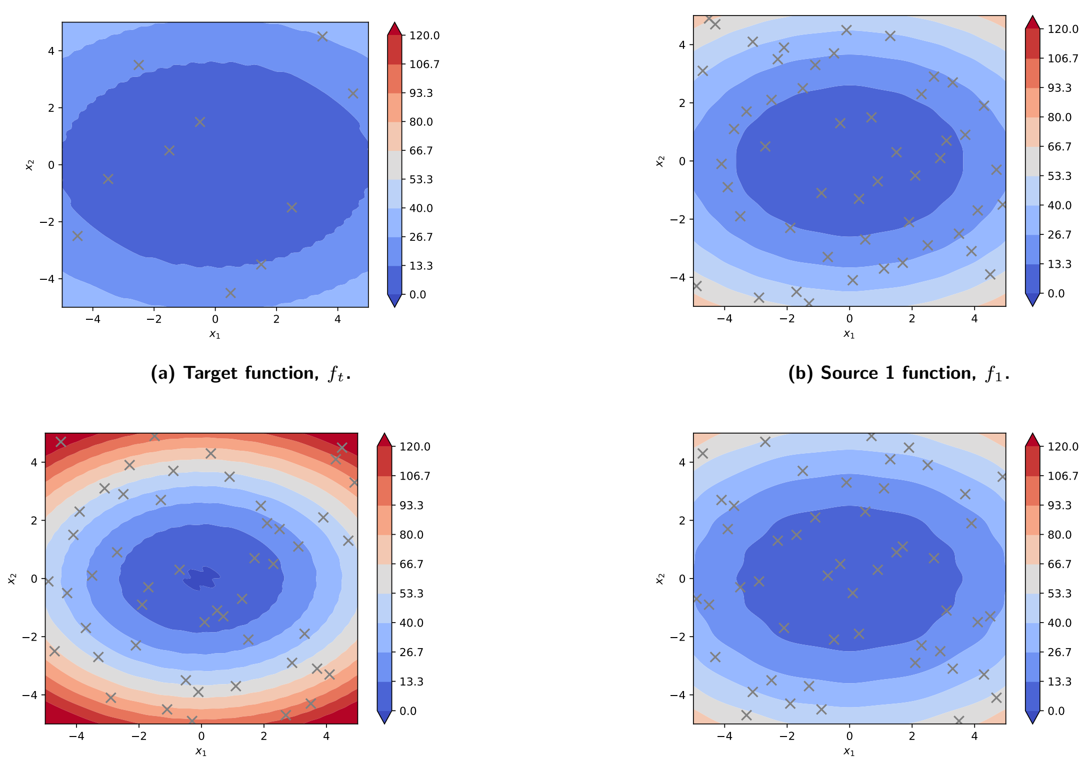
$$f_1(x_1, x_2) = x_1^2 + 2x_2^2 - 0.3\cos(\pi x_1) - 0.4\cos(2\pi x_2) + 0.7, \tag{27}$$
$$f_2(x_1, x_2) = x_1^2 + 2x_2^2 - 0.3\cos(\pi x_1)\cos(2\pi x_2), \tag{28}$$
$$f_3(x_1, x_2) = 2x_1^2 + 4x_2^2 - 0.3\cos(3\pi x_1 + 4\pi x_2) - 0.5, \tag{29}$$

and

$$f_t(x_1, x_2) = 0.5x_1^2 + x_2^2 - 0.3\cos(3\pi x_1 + 4\pi x_2) + 0.4. \tag{30}$$

Each source function was used to create a DOE from the same design space as the target problem design space such that $x_1 \in [-5, 5]^\top$, $x_2 \in [-5, 5]^\top$. A low number of DOE evaluations is selected for the target problem to demonstrate the ability of the transfer learning to adjust model priors and select source surrogate models with limited target data. In our tests, we use a DOE of size 10 for the target problem (f_t) and a DOE of 50 for each of the three source problems (i.e., f_1, f_2 and f_3).

We first perform the DOE evaluation of all the problems as presented in Figure 7. Algorithm 1 is then used to obtain the modified source models $\hat{y}^1_{\mathrm{S}'}, \hat{y}^2_{\mathrm{S}'}$ and $\hat{y}^3_{\mathrm{S}'}$. Since we are interested to demonstrate



(a) Target function, f_t. (b) Source 1 function, f_1. (c) Source 2 function, f_2. (d) Source 3 function, f_3.

Figure 7: Contour plots of the target and source functions showing the DOE evaluation points corresponding to each problem.

the prediction accuracy in this illustration example, we use fixed criteria weights in Equation (20) by setting a value of 1 to the weight of accuracy criterion. We do not assign any values for the weight of shape criterion since we know that all the studied functions in Equations (27–30) share a similar bowl shape. Similarly, we do not assign any values for the weight of the variance criterion since the source models are all built on a *LHS* sampling of the same design space. We set the number of clusters in this illustration example to one since all the source models behave similarly across the whole design space. We then use Algorithm 3 to obtain the ensemble of surrogates model. We create 20 random evaluations in the design space of the target problem and use them as test data $\text{DoE}_{test} : (X_{test}, Y_{test})$.

We test the following methods in this illustration problem: the target model $(\hat{y}_{\mathbf{t}})$, the source surrogate models $(\hat{y}_{\mathbf{S}}^{1,\dots,N})$, the modified source surrogate models $(\hat{y}_{\mathbf{S'}}^{1,\dots,N})$, and the ensemble of surrogates without clustering (Ensemble-TL). All the tested surrogate models, built solely on the 10-evaluation DoE_T, are then used to estimate f_t based on the input vector of the test data X_{test}. Figure 8 presents the results comparing the prediction errors of the tested models showing the benefits of transfer learning in adjusting the source models and correctly creating an ensemble of surrogates. The x-axis lists all the models that are compared for predicting the 50 random test evaluations in the full design space of the target problem. The y-axis presents box plots showing the errors of each model at the 20 tested evaluations. We note that all modified source models, $\hat{y}^1_{\text{S}'}$, $\hat{y}^2_{\text{S}'}$ and $\hat{y}^3_{\text{S}'}$, show lower prediction errors than the target model $\hat{y}_{\text{t}}$ even if they are built on inaccurate source models, $\hat{y}^1_{\text{S}}$, $\hat{y}^2_{\text{S}}$ and $\hat{y}^3_{\text{S}}$. In addition, the ensemble of surrogates prediction errors show that the ensemble probability weighting algorithm from Equation (23) is able to select the modified source models that maximize accuracy.

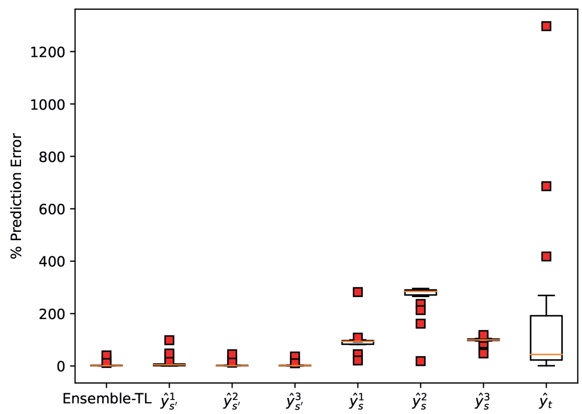


Figure 8: Box plot comparing ensemble of surrogates with transfer learning using prior with model predictions of the modified sources and target at 20 test evaluations.

We also compare the ensemble of surrogates without clustering (Ensemble-TL) with existing state-of-the-art methods: Two-Stage Transfer surrogate model with Rankings (TST-R) from [60] and Ranking-weighted Gaussian Process Ensemble (RGPE) from [16]. We present box plots of the predictions in Figure 9 noting the capability of our methods of predicting accurate function evaluations. Results of TST-R and RGPE are expected since both of these methods rely on the shape of source functions for target predictions.

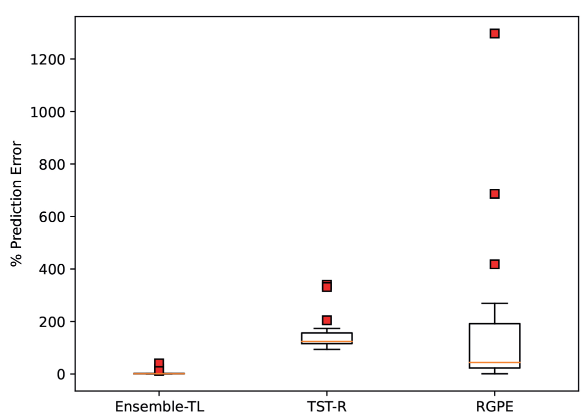


Figure 9: Box plot comparing ensemble of surrogates with existing state-of-the-art methods using 20 test evaluations..

4.2.2 Modified Giunta functions

For the second analytical problem, we aim to demonstrate the capability of the mixture of experts algorithm combined with the ensemble of surrogates. We use the Giunta function [25], a continuous multi-modal two-dimensional function, as the target function $f_t(x)$.

$$f_t(x) = 0.6 + \sum_{i=1}^{2} \left[\sin(\frac{16}{15}x_i - 1) + \sin^2(\frac{16}{15}x_i - 1) + \frac{1}{50}\sin(4(\frac{16}{15}x_i - 1)) \right]$$

subject to $-1 \leqslant x_i \leqslant 1$ for $i \in [1, 2]$. The goal of this analytical problem is to show that Algorithm 3 correctly clusters the design space based on outputs of a set of source surrogate models and then calculates the probabilities of the ensemble of surrogates in each of the identified clusters. We define two source functions, $f_1(x)$ and $f_2(x)$ that are equal to the target function $f_t(x)$ only in specific regions of the design space such that:

$$f_1(x) = \begin{cases} f_t(x) & \text{if } x_2 > 0.5 \\ f_t(x) + 3x_2^2 & \text{otherwise,} \end{cases}$$

and

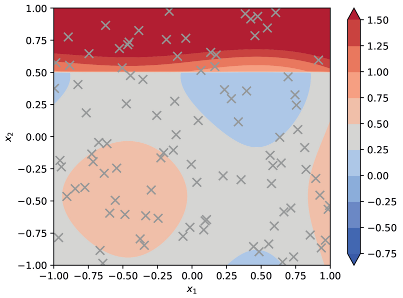
$$f_2(x) = \begin{cases} f_t(x) & \text{if } x_2 < 0.5, \\ f_t(x) - 3x_2^2 & \text{otherwise.} \end{cases}$$

We create a DOE of 100 evaluations for each of the source problems and 30 evaluations for the target problem. We set the number of clusters to 3 which is the minimum clusters needed to properly separate the different regions of the design space. We run Algorithm 3 to produce the clusters K and the ensemble of surrogates without modifying the source surrogate models, i.e., $\hat{y}^n_{s'} = \hat{y}^n_s$. Figure 10 shows contour plots of the two source functions, $f_1(x)$ and $f_2(x)$, the target function $f_t(x)$, and their associated function evaluations, $\mathrm{DoE}^1_{\mathrm{S}}, \mathrm{DoE}^2_{\mathrm{S}}$ and $\mathrm{DoE}_{\mathrm{T}}$.After the ensemble of surrogates is built, we randomly select 100 test points from the full design space $\mathrm{DoE}_{test} : (\mathrm{X}_{test}, \mathrm{Y}_{test})$, divide the points by the clusters K, and apply the source surrogate models, $(\hat{y}^1_s)$ and $(\hat{y}^2_s)$, the ensemble of surrogates without clustering (Ensemble-TL), the ensemble of surrogates with *smooth* clustering and prediction (Ensemble Smooth-C), and the ensemble of surrogates with *hard* clustering and prediction (Ensemble Hard-C).

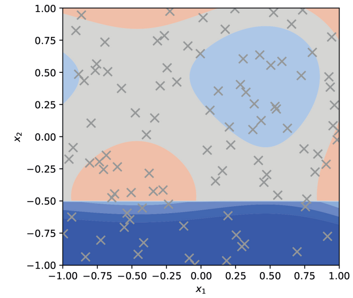
We show box plots of the methods tested in each of the clusters and in the entire design space in Figure 11. We note that (Ensemble Hard-C) shows the lowest prediction errors on the test data in all the clusters, as is expected in problems where there is an abrupt transition between the clusters. The behavior of the source surrogates $(\hat{y}^1_s)$ and $(\hat{y}^2_s)$ depends on the cluster selected as expected where $(\hat{y}^1_s)$ shows low prediction errors in cluster 1 in Figure 11 (b) whereas $(\hat{y}^2_s)$ shows low prediction errors in cluster 3 in Figure 11 (c). (Ensemble-TL) results show the that method balances the between $(\hat{y}^1_s)$ and $(\hat{y}^2_s)$ across the whole design space which leads to higher prediction errors than (Ensemble Hard-C) and (Ensemble Smooth-C).

We present in Figure 12 contour plots of (Ensemble Hard-C) and (Ensemble Smooth-C) across the design space. We note the behavior of (Ensemble Smooth-C) in cluster 3 ($x_2 > 0.5$) is a combination of the shapes of the surrogate models from Source 1 and Source 2 which explains the higher prediction errors of (Ensemble Smooth-C) in Figure 11 (d). Whereas (Ensemble Hard-C) is based solely on the surrogate model of Source 2 in cluster 3 and consequently has the lowest prediction errors.

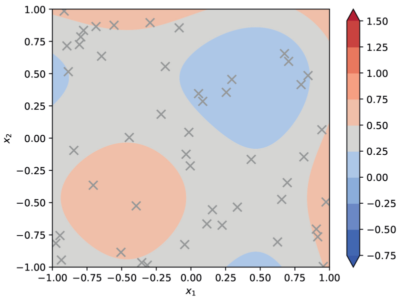
The above results are based on equal weights for the shape, accuracy, and variance criteria with equal bandwidths set at 1 assuming that you have no prior knowledge about the transfer learning problem. We perform the hyperparameter optimization formulation in Equation (26) to obtain the optimal criteria weights and bandwidths for this problem using (Ensemble Smooth-C). The minimum errors achieved were achieved when only the accuracy criterion is used and its bandwidth is set to 0.1. We also perform a parametric analysis for the criteria weights and bandwidths hyperparameters that influence the behavior of the (Ensemble Smooth-C) algorithm.



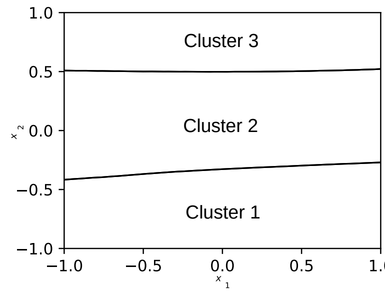
(a) **Source 1 function, f_1, contour plot and DOE evaluations.**



(b) **Source 2 function, f_2, contour plot and DOE evaluations.**

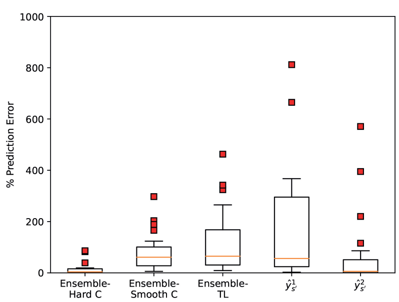


(c) **Target function, f_t, contour plot and DOE evaluations.**

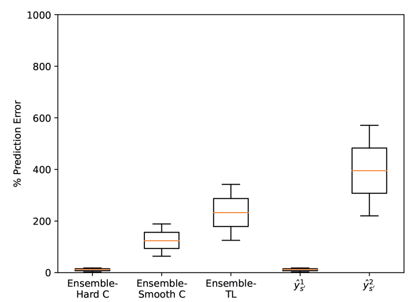


(d) **Clusters definition and separation lines.**

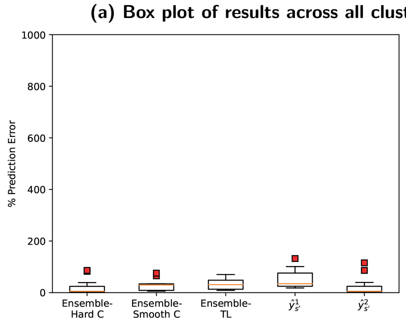
Figure 10: Giunta problem illustration contour plots of the source and target functions showing the DOE evaluation points corresponding to each problem.



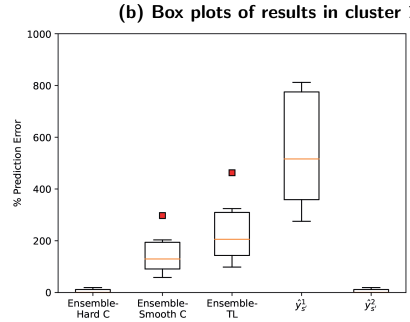
(a) **Box plot of results across all clusters.**



(b) **Box plots of results in cluster 1.**



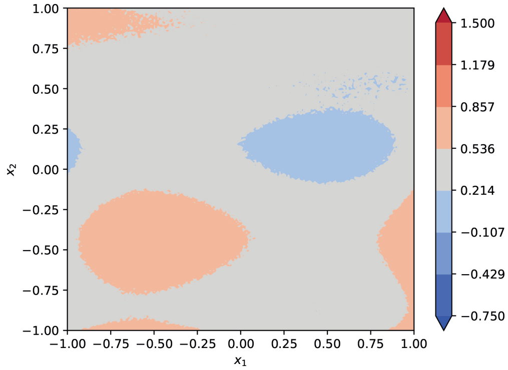
(c) **Box plots of results in cluster 2.**



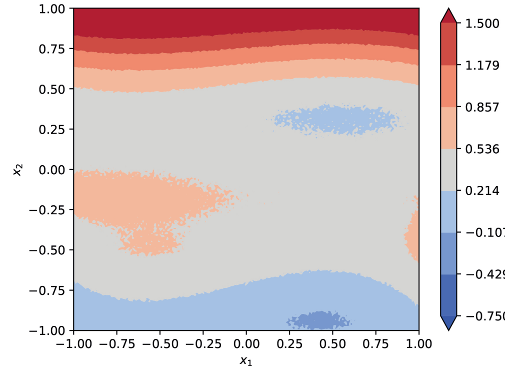
(d) **Box plots of results in cluster 3.**

Figure 11: Box plots comparing the results of the different surrogate models and ensemble of surrogates tested on 20 evaluation tests and showing (Ensemble Hard-C) as the best performing method across all clusters.

In Figure 13, we compare the results using either the accuracy, shape, or variance criteria. We note that using the variance criterion solely is not ideal for this type of problems especially that all source data is obtained from the same design space. The accuracy criterion produces the lowest prediction errors. However, we are aware that based on the weighting of the ensemble of surrogates solely on accuracy may yield inaccurate results in some problems as discussed in [16].



(a) Ensemble Hard-C



(b) Ensemble Smooth-C

Figure 12: Contour plots using (Ensemble Hard-C) in (a) and (Ensemble Smooth-C) in (b).

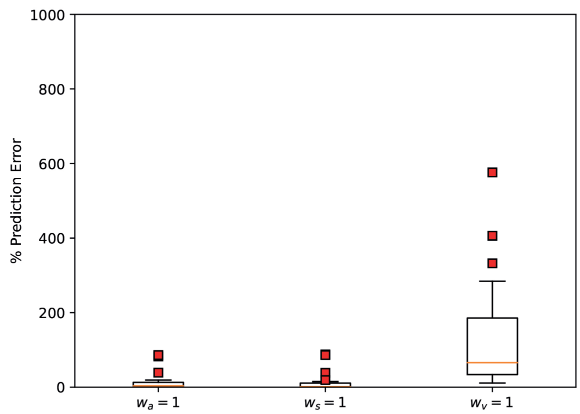


Figure 13: Box plots of results using either the accuracy w_a, shape w_s, or variance w_v criteria using 20 test evaluations.

In Figure 14, we compare the results using different set values for the accuracy bandwidth ρ_a when using only the accuracy criterion showing the importance of selecting appropriate bandwidths for each problem. The results so far are based on the Epanechnikov kernel in Equation (18). In Figure 15, we study the impact of kernel selection on prediction errors. We select two popular kernels [35]: the triweight kernel, denoted as **TRI-W**, and the Gaussian kernel, denoted as **GAUS**. The results show no significant difference in prediction errors when compared to the baseline Epanechnikov kernel, denoted as **EPAN**. Therefore, we use the Epanechnikov kernel for the remainder of this work. It is noted that the results presented are dependent on the considered DOE as they are based on a single run.

We also compare our proposed methods with existing state-of-the-art methods: (TST-R) from [60] and (RGPE) from [16]. We present box plots of the predictions across all clusters in Figure 16 noting the capability of (Ensemble Hard-C) of predicting accurate function evaluations.

4.3 Aircraft conceptual design problems

We use an aircraft conceptual design problem to demonstrate our transfer learning approach. We present two different scenarios based on the problems defined in Section 2:

Scenario 1: heterogeneous output transfer learning, i.e., source problems that do not share the same output as the target problem.

Scenario 2: surrogate modeling transfer learning across regions of the design space, i.e., source surrogate models that do not correlate to the target problem similarly across the whole defined design space.

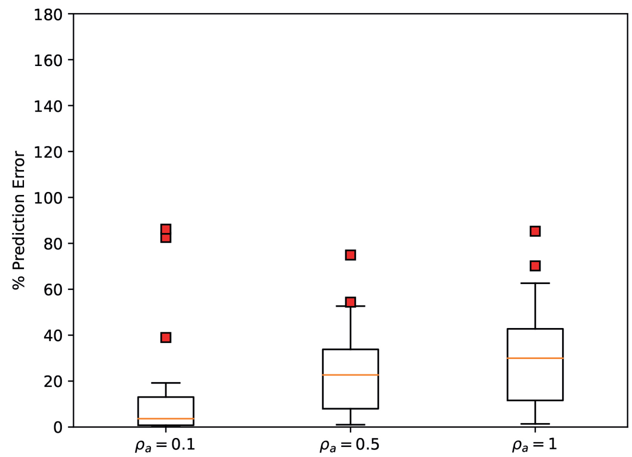


Figure 14: Box plots of results using different set values for the accuracy bandwidth ρ_a using 20 test evaluations.

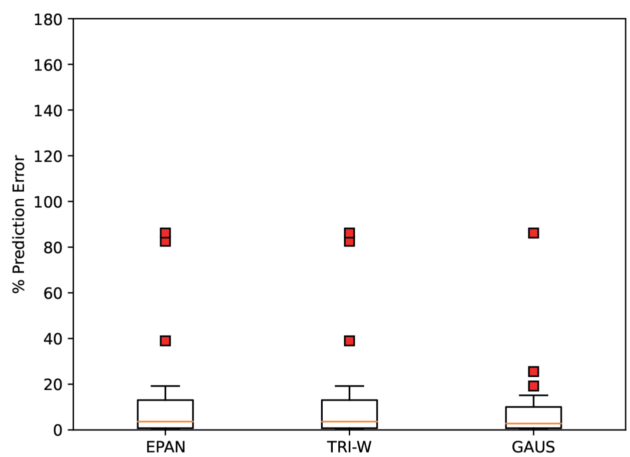


Figure 15: Box plots of results using different kernel selection using 20 test evaluations.

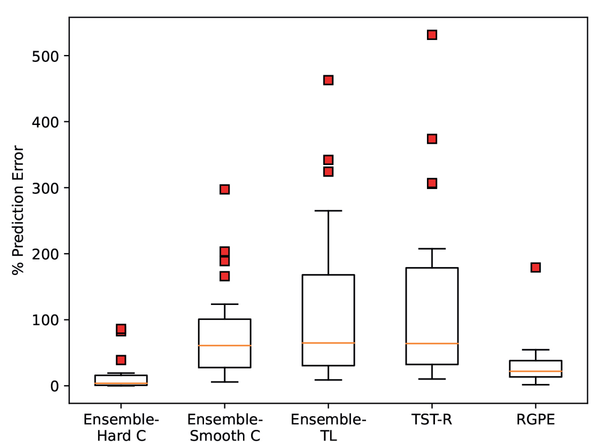


Figure 16: Box plot comparing our ensemble surrogate models with existing state-of-the-art methods using 20 test evaluations.

The first scenario is based on the aircraft noise prediction problem presented in Section 2.1. We extend the problem by considering four sets of source data that have different outputs from a target problem while assuming a relationship exists between each of the source problems' outputs and the target problem output. There are numerous cases where designers are interested in a certain parameter but the models needed to produce this parameter are either unavailable or computationally expensive. As we describe in Section 2.1, we are interested in building a noise prediction model based on a limited amount of data from an aircraft noise simulation code based on an input vector x. We intend to use abundantly available data that link these same inputs with aircraft MTOW assuming that the behavior of MTOW and noise are similar in response to changes in x. Aircraft block fuel consumption is the total amount of fuel an aircraft burns throughout its design mission. Similarly, aircraft CO_2 emissions are the total emitted CO_2 throughout the design mission of the aircraft. Although, a correlation between aircraft approach noise and either block fuel or CO_2 emissions is not guaranteed since an aircraft's range, a major contributor to the block fuel and CO_2 emissions, is not correlated with aircraft noise, we still use these two parameters in the source data to test how our methods are able to select which source problem is relevant to the predictions of the target problem. Aircraft cost is also another parameter that we use to test our methods although no obvious relationship exist between cost and noise. The setup of the source and target problems is presented in Table 2. We consider that each set of source data has a DOE of 200 evaluations; whereas, the target problem has a smaller DOE of only 10 evaluations. We use *LHS* as the sampling method for all DOE's in this scenario. Although the input variables of the source and target models are not identical, we use an aircraft level multidisciplinary analysis framework [42, 43, 53] to obtain a common set of inputs and design space that generate the corresponding inputs needed for each of the source and target problems as defined in Table 3. Algorithm 3 is then used to obtain the modified source models and ensemble of surrogates. In this scenario, we are interested in obtaining prediction accuracy in addition to ensuring that the relationships of the target input-output are captured. Therefore, we use fixed criteria weights in Equation (20) by setting equal values of 1/3 to the weights of shape, accuracy, and variance criteria. We assign the number of clusters to 1 in this scenario to focus this problem only on the model scaling and ensemble weighting parts of Algorithm 3.

Table 2: Setup of source and target problems in aircraft design scenario 1.

Problem	Output	DOE Size
Source 1	Aircraft Block Fuel Consumption	200
Source 2	Aircraft CO_2 Emissions	200
Source 3	Aircraft Cost	200
Source 4	Aircraft MTOW	200
Target	Aircraft Approach Noise	10

Table 3: A list of the source and target model inputs related to the aircraft conceptual design problems.

Input	Description
x_1	Rubber engine scaling factor
x_2	Rubber engine bypass ratio
x_3	Rubber engine overall pressure ratio
x_4	Wing aspect ratio
x_5	Wing area
x_6	Wing trailing edge sweep
x_7 and x_8	Wing rear spar chord-wise location
x_9	Wing sweep
x_{10}	Wing taper ratio
$x_{11}, \ldots, x_{14}$	Wing thickness-to-chord ratios

For the test data, we create 50 additional evaluations of the target model and use these evaluations to test the developed models. As shown in Figure 17, the proposed algorithm shows lower prediction errors against the test data when compared to the target model $\hat{y}_{\mathrm{t}}$ that is solely built on the target

DOE. We also note that the method is able to only select the source models that correlate to the target problem as source models associated with block fuel consumption and CO_2 emissions were not selected. Also, an interesting outcome shows that the cost source model results in low prediction errors that can be explained due to the relationship between aircraft cost and MTOW, which aircraft noise correlates to.

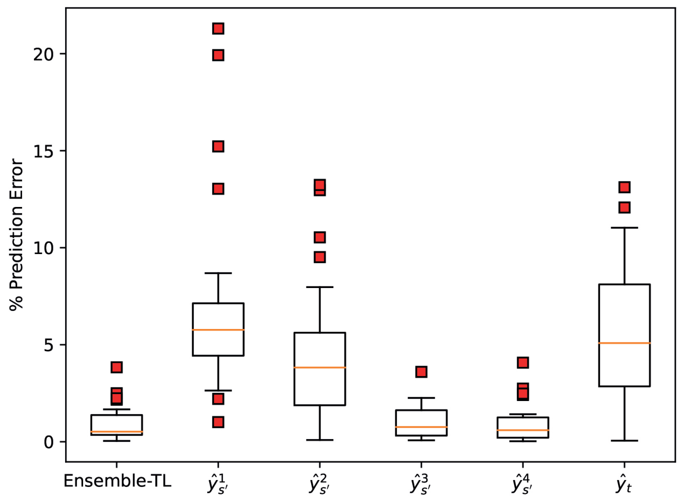


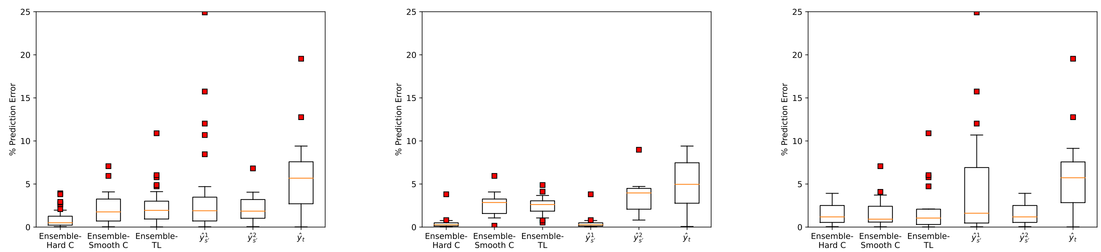
Figure 17: Box plot comparing (Ensemble-TL) with model predictions of the modified sources and target of aircraft design scenario 1.

The second scenario is based on the problem defined in Section 2.2 where the problem of interest is the prediction of MTOW of an aircraft depending on whether it is slatted or unslatted. In this scenario, we are interested in performing analyses that predict the MTOW of an aircraft based on typical aircraft sizing variables from Table 3. Therefore, we append the design space of Table 3 to include a discrete input, $x_{15} \in \{0, 1\}$, that defines whether slats are used on the aircraft or not. As presented in Section 2.2, we use two source problems that are obtained using two different models: the first is able to properly model the MTOW for aircraft with slats, and the second is able to properly model the MTOW for unslatted aircraft. The scenario DOE setup is presented in Table 4. Each source problem is used to model their corresponding output for the majority of the DOE with the exception of a small number of evaluations that use the opposing slat discrete input variable value. The latter DOE evaluations are expected to yield incorrect results compared to their respective correct models in regions of the design space where x_{15} does not match the source problem. The target problem uses correct models for both the slatted and unslatted aircraft configurations. Therefore, in the target problem we expect that surrogate models obtained from the slatted aircraft source problem produce accurate results in the region of the design space where $x_{15} = 1$ and inaccurate results in the region of the design space where $x_{15} = 0$; the opposite assumption is expected for surrogates obtained from the unslatted aircraft source problem.

Table 4: Setup of source and target problems in aircraft design scenario 2.

Problem	DOE Size with Slats	DOE Size without Slats
Source 1 (slatted)	100	10
Source 2 (unslatted)	10	100
Target	10	10

We set the number of clusters in this scenario to two which is equivalent to the split design space of x_{15}. Similar to scenario 1, we set equal values of 1/3 to the weights of shape, accuracy, and variance criteria to be used in Equation (20). We create a test dataset from the target problem containing both slatted and unslatted aircraft and use it to validate the results of the ensemble of surrogates. We note from the presented results in Figure 18 that (Ensemble Hard-C) performs best compared to other ensemble models and the source and target models across the whole design space similar to the results of the second numerical example. Although the motivations and application cases presented in this work focus on aircraft design, the transfer learning methods presented can also be beneficial for other domains. Our methods can be applied to any target discipline where generating data is computationally expensive and other existing data that can be correlated to this discipline is available.



(a) Results in full design space **(b) Results in cluster 1 (slatted)** **(c) Results in cluster 2 (unslatted)**

Figure 18: Box plots comparing the results of the ensemble of surrogates, modified source surrogate models, and target surrogate model.

5 Conclusions

In this paper, we presented a methodology to perform transfer learning between a set of source problems and a target problem using an ensemble of surrogates in a divided design space. Our methodology first creates modified surrogate models of the source problems fitted to the target data using the posteriors of the source surrogate models. We then divide the design space of the target problem using the clustering EM algorithm in order to create a mixture of experts in each cluster. Finally, we developed a multi-criteria weighting approach to build the ensemble of surrogates based on shape, accuracy, and variance of the surrogates. We validated the proposed methodology using two scenarios of aircraft conceptual design problems by performing a comparative analysis of prediction errors of the ensemble of surrogates and the target surrogate models without transfer learning. We demonstrate that the transfer learning approach is able to capture the behavior of source models and adjust it to the target problem showing significant improvements in prediction errors (up to 10%). Our proposed multi-criteria ensemble weighting approach provides users tuning parameters that adjust how information is transferred and can be adapted based on the studied problem. Since the presented ensemble of surrogate models provides information for both mean and variance predictions, it can be also employed in the context of constrained Bayesian optimization to speed up the convergence. In fact, in such scenarios, aircraft designers are interested in creating surrogate models of objective and constraint functions in Bayesian optimization based on previous optimization results. We note however that using the transfer learning algorithms presented herein at every Bayesian optimization iteration may significantly increase the computational cost. Methods to address this potential computational overhead will be investigated in future work.

References

[1] John Aldrich. Ra fisher and the making of maximum likelihood 1912-1922. Statistical science, 12(3):162–176, 1997.

[2] Mauricio A. Alvarez, Lorenzo Rosasco, Neil D. Lawrence, et al. Kernels for vector-valued functions: A review. Foundations and Trends in Machine Learning, 4(3):195–266, 2012.

[3] Malek Ben Salem and Lionel Tomaso. Automatic selection for general surrogate models. Structural and Multidisciplinary Optimization, 58(2):719–734, 2018.

[4] Dimitri Bettebghor, Nathalie Bartoli, Stéphane Grihon, Joseph Morlier, and Manuel Samuelides. Surrogate modeling approximation using a mixture of experts based on em joint estimation. Structural and Multidisciplinary Optimization, 43:243–259, 2011.

[5] Ihor O. Bohachevsky, Mark E. Johnson, and Myron L. Stein. Generalized simulated annealing for function optimization. Technometrics, 28(3):209–217, 1986.

[6] Steven L. Brunton, J. Nathan Kutz, Krithika Manohar, Aleksandr Y. Aravkin, Kristi Morgansen, Jennifer Klemisch, Nicholas Goebel, James Buttrick, Jeffrey Poskin, Adriana W. Blom-Schieber, et al. Data-driven aerospace engineering: reframing the industry with machine learning. AIAA Journal, 59(8):2820–2847, 2021.

[7] Kenneth P. Burnham and David R. Anderson. Multimodel inference: understanding AIC and BIC in model selection. Sociological Methods & Research, 33(2):261–304, 2004.

[8] Jasper H. Bussemaker, Paul Saves, Nathalie Bartoli, and et al. Surrogate-based optimization of system architectures subject to hidden constraints. In AIAA Aviation Forum and ASCEND, pages AIAA 2024–4401, 2024.

[9] Yashovardhan S. Chati and Hamsa Balakrishnan. Modeling of aircraft takeoff weight using gaussian processes. Journal of Air Transportation, 26(2):70–79, 2018.

[10] Hao Chen, Zhilang Zhang, Weikun Li, Qimeng Liu, Kai Sun, Dixia Fan, and Weicheng Cui. Ensemble of surrogates in black-box-type engineering optimization: Recent advances and applications. Expert Systems with Applications, page 123427, 2024.

[11] Richard Curran, Mark Price, Srinivasan Raghunathan, Emmanuel Benard, Stephen Crosby, Sylvie Castagne, and Paul Mawhinney. Integrating aircraft cost modeling into conceptual design. Concurrent Engineering, 13:321–330, 2005.

[12] Andreas Damianou and Neil D. Lawrence. Deep Gaussian processes. In Artificial intelligence and statistics, pages 207–215. PMLR, 2013.

[13] Arthur P. Dempster, Nan M. Laird, and Donald B. Rubin. Maximum likelihood from incomplete data via the em algorithm. Journal of the royal statistical society: series B (methodological), 39(1):1–22, 1977.

[14] Lixin Duan, Ivor W. Tsang, and Dong Xu. Domain transfer multiple kernel learning. IEEE Transactions on Pattern Analysis and Machine Intelligence, 34(3):465–479, 2012.

[15] Vassiliy A. Epanechnikov. Non-parametric estimation of a multivariate probability density. Theory of Probability & Its Applications, 14(1):153–158, 1969.

[16] Matthias Feurer, Benjamin Letham, and Eytan Bakshy. Scalable meta-learning for Bayesian optimization using ranking-weighted gaussian process ensembles. In AutoML Workshop at ICML, volume 7, 2018.

[17] Matthias Feurer, Benjamin Letham, Frank Hutter, and Eytan Bakshy. Practical transfer learning for Bayesian optimization. arXiv:1802.02219, 2018.

[18] Antonio Filippone. Aircraft Noise Prediction. Progress in Aerospace Sciences, 68:27–63, 2014.

[19] Alexander Forrester, Andras Sobester, and Andy Keane. Engineering design via surrogate modelling: a practical guide. John Wiley & Sons, New Jersey, NY, 2008.

[20] Jacob R. Gardner, Matt J. Kusner, Zhixiang Eddie Xu, Kilian Q. Weinberger, and John P Cunningham. Bayesian optimization with inequality constraints. In ICML, volume 2014, pages 937–945, 2014.

[21] Ana Garcia Garriga, Laura Mainini, and Sangeeth Saagar Ponnusamy. A machine learning enabled multi-fidelity platform for the integrated design of aircraft systems. Journal of Mechanical Design, 141(12):121405, 2019.

[22] Tushar Goel, Raphael T. Haftka, Wei Shyy, and Nestor V. Queipo. Ensemble of surrogates. Structural and Multidisciplinary Optimization, 33:199–216, 2007.

[23] Snorri Gudmundsson. General Aviation Aircraft Design: Applied Methods and Procedures. Butterworth-Heinemann, Oxford, United Kingdom, 2013.

[24] Ryan P. Henderson, Joaquim R.R.A. Martins, and Ruben E. Perez. Aircraft conceptual design for optimal environmental performance. The Aeronautical Journal, 116:1–22, 2012.

[25] Momin Jamil and Xin-She Yang. A literature survey of benchmark functions for global optimisation problems. International Journal of Mathematical Modelling and Numerical Optimisation, 4(2):150–194, 2013.

[26] Momin Jamil and Xin-She Yang. A literature survey of benchmark functions for global optimization problems. CoRR, abs/1308.4008, 2013.

[27] Michael I. Jordan and Robert A. Jacobs. Hierarchical mixtures of experts and the em algorithm. Neural Computation, 6(2):181–214, 1994.

[28] Melih Kandemir. Asymmetric transfer learning with deep Gaussian processes. In International Conference on Machine Learning, pages 730–738. PMLR, 2015.

[29] Maurice G. Kendall. A new measure of rank correlation. Biometrika, 30(1-2):81–93, 1938.

[30] Rui Leite, Pavel Brazdil, and Joaquin Vanschoren. Selecting classification algorithms with active testing. In Machine Learning and Data Mining in Pattern Recognition: 8th International Conference, MLDM 2012, Berlin, Germany, July 13-20, 2012. Proceedings 8, pages 117–131. Springer, 2012.

[31] Rhea P. Liem, Charles A. Mader, and Joaquim R.R.A. Martins. Surrogate models and mixtures of experts in aerodynamic performance prediction for aircraft mission analysis. Aerospace Science and Technology, 43:126–151, 2015.

[32] Marius Lindauer, Katharina Eggensperger, Matthias Feurer, André Biedenkapp, Difan Deng, Carolin Benjamins, Tim Ruhkopf, René Sass, and Frank Hutter. SMAC3: A versatile Bayesian optimization package for hyperparameter optimization. Journal of Machine Learning Research, 23(54):1–9, 2022.

[33] Marius Lindauer and Frank Hutter. Warmstarting of model-based algorithm configuration. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 32, 2018.

[34] X Liu, Q Zhu, and H Lu. Modeling multiresponse surfaces for airfoil design with multiple-output-gaussian-process regression. Journal of Aircraft, 51(3):740–747, 2014.

[35] James S. Marron and Deborah Nolan. Canonical kernels for density estimation. Statistics & Probability Letters, 7(3):195–199, 1988.

[36] Zvika Marx, Michael T. Rosenstein, Leslie P. Kaelbling, and Thomas G. Dietterich. Transfer learning with an ensemble of background tasks. Inductive Transfer, 10, 2005.

[37] Geoffrey J. McLachlan and Thriyambakam Krishnan. The EM Algorithm and Extensions. John Wiley & Sons, Hoboken, NJ, 2007.

[38] Alan Tan Wei Min, Abhishek Gupta, and Yew-Soon Ong. Generalizing transfer Bayesian optimization to source-target heterogeneity. IEEE Transactions on Automation Science and Engineering, 18(4):1754–1765, 2020.

[39] Alan Tan Wei Min, Ramon Sagarna, Abhishek Gupta, Yew-Soon Ong, and Chi Keong Goh. Knowledge transfer through machine learning in aircraft design. IEEE Computational Intelligence Magazine, 12(4):48–60, 2017.

[40] Pablo Moreno-Muñoz, Antonio Artés, and Mauricio Alvarez. Modular Gaussian processes for transfer learning. Advances in Neural Information Processing Systems, 34:24730–24740, 2021.

[41] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, and et al. Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12:2825–2830, 2011.

[42] Pat Piperni, Antoine DeBlois, and Ryan Henderson. Development of a multilevel multidisciplinary-optimization capability for an industrial environment. AIAA journal, 51:2335–2352, 2013.

[43] Remy Priem, Hugo Gagnon, Ian Chittick, Stephane Dufresne, Youssef Diouane, and Nathalie Bartoli. An efficient application of Bayesian optimization to an industrial MDO framework for aircraft design. AIAA Aviation 2020 Forum, pages AIAA 2020–3152, 2020.

[44] Ashwin Renganathan, Romit Maulik, and Jai Ahuja. Enhanced data efficiency using deep neural networks and Gaussian processes for aerodynamic design optimization. Aerospace Science and Technology, 111:106522, 2021.

[45] Peter K.C. Rudolph. High-lift systems on commercial subsonic airliners. Technical report, National Aeronautics and Space Administration, 1996.

[46] Mohammad H. Sadraey. Aircraft Design: a Systems Engineering Approach. John Wiley & Sons, New Hampshire, 2012.

[47] P. Saves, R. Lafage, N. Bartoli, Y. Diouane, and et al. SMT 2.0: A Surrogate Modeling Toolbox with a focus on Hierarchical and Mixed Variables Gaussian Processes. Advances in Engineering Software, 188:103571, 2024.

[48] Paul Saves, Nathalie Bartoli, Youssef Diouane, Thierry Lefebvre, Joseph Morlier, Christophe David, Eric Nguyen Van, and Sébastien Defoort. Bayesian optimization for mixed variables using an adaptive dimension reduction process: applications to aircraft design. In AIAA SciTech 2022 Forum, pages AIAA 2022–0082, 2022.

[49] Grigorios Skolidis. Transfer learning with Gaussian processes. PhD thesis, The University of Edinburgh, 2012.

[50] Takashi Takekawa. Clustering of non-gaussian data by variational bayes for normal inverse gaussian mixture models. arXiv preprint arXiv:2009.06002, 2020.

[51] Chuanqi Tan, Fuchun Sun, Tao Kong, Wenchang Zhang, Chao Yang, and Chunfang Liu. A survey on deep transfer learning. In Artificial Neural Networks and Machine Learning–ICANN 2018: 27th International Conference on Artificial Neural Networks, Rhodes, Greece, October 4-7, 2018, Proceedings, Part III 27, pages 270–279. Springer, 2018.

[52] Timothy Tay and Carolina Osorio. Bayesian optimization techniques for high-dimensional simulation-based transportation problems. Transportation Research Part B: Methodological, 164:210–243, 2022.

[53] Ali Tfaily, Youssef Diouane, Nathalie Bartoli, and Michael Kokkolaras. Bayesian optimization with hidden constraints for aircraft design. Structural and Multidisciplinary Optimization, 67(7):123, 2024.

[54] Ali Tfaily and Michael Kokkolaras. Integrating air systems in aircraft multidisciplinary design optimization. In 2018 Multidisciplinary Analysis and Optimization Conference, pages AIAA 2018–3742, 2018.

[55] David J.J. Toal and Andy J. Keane. Efficient multipoint aerodynamic design optimization via cokriging. Journal of Aircraft, 48(5):1685–1695, 2011.

[56] Michael T. Tong. Using machine learning to predict core sizes of high-efficiency turbofan engines. Journal of Engineering for Gas Turbines and Power, 141(11):111023, 2019.

[57] Karl Weiss, Taghi M Khoshgoftaar, and DingDing Wang. A survey of transfer learning. Journal of Big Data, 3:1–40, 2016.

[58] Christopher Williams and Carl Rasmussen. Gaussian processes for machine learning. MIT press, Cambridge, MA, 2006.

[59] Martin Wistuba and Josif Grabocka. Few-shot Bayesian optimization with deep kernel surrogates. arXiv:2101.07667, 2021.

[60] Martin Wistuba, Nicolas Schilling, and Lars Schmidt-Thieme. Two-stage transfer surrogate model for automatic hyperparameter optimization. In Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2016, Riva del Garda, Italy, September 19-23, 2016, Proceedings, Part I 16, pages 199–214. Springer, 2016.

[61] Rui Xia, Wessel Bruinsma, William Tebbutt, and Richard E Turner. The gaussian process latent autoregressive model. In Third Symposium on Advances in Approximate Bayesian Inference, 2020.

[62] Pengcheng Ye and Guang Pan. Selecting the best quantity and variety of surrogates for an ensemble model. Mathematics, 8(10):1721, 2020.

[63] Dani Yogatama and Gideon Mann. Efficient transfer learning method for automatic hyperparameter tuning. In Artificial intelligence and statistics, pages 1077–1085. PMLR, 2014.

[64] Byoung Hyun Yoo, Junhwan Kim, Byun-Woo Lee, Gerrit Hoogenboom, and Kwang Soo Kim. A surrogate weighted mean ensemble method to reduce the uncertainty at a regional scale for the calculation of potential evapotranspiration. Scientific Reports, 10(1):870, 2020.

[65] Oleksandr Zaporozhets, Vadim Tokarev, and Keith Attenborough. Aircraft Noise: Assessment, Prediction and Control. CRC Press, Boca Raton, FL, 2011.

[66] Qingzhi Zhong, Huazhen Lin, and Yi Li. Cluster non-gaussian functional data. Biometrics, 77(3):852–865, 08 2020.

[67] Fuzhen Zhuang, Zhiyuan Qi, Keyu Duan, Dongbo Xi, Yongchun Zhu, Hengshu Zhu, Hui Xiong, and Qing He. A comprehensive survey on transfer learning. Proceedings of the IEEE, 109(1):43–76, 2020.