33RD CONGRESS
OF THE INTERNATIONAL COUNCIL
OF THE AERONAUTICAL SCIENCES
STOCKHOLM, SWEDEN, 4-9 SEPTEMBER, 2022

ICAS
2022
SWEDEN

AIRCRAFT DESIGN PARAMETER ESTIMATION USING DATA-DRIVEN MACHINE LEARNING MODELS

Sooryun Shin[1], Sanga Lee[1], Chankyu Son[2], Kwanjung Yee[3]

[1]Korea Institute of Industrial Technology, Cheonan, Korea
[2]Cheongju University, Cheongju, Korea
[3]Seoul National University, Seoul, Korea

Abstract

Selecting an appropriate set of the initial parameter for the aircraft conceptual design phase has a huge effect on the overall design result of the aircraft. Therefore, estimating appropriate combinations of the initial design parameters is an essential step for obtaining feasible conceptual design results. Existing methods selected the initial parameters by using empirical formulas or adopting the empirical parameters. These methods have limitations in that the design results are highly dependent on the user's experience and prior knowledge. Moreover, it is necessary to evaluate numerous empirical formulas to figure out the relationships between diverse combinations of variables. To overcome these limitations, this study applied data-driven machine learning techniques. The data-driven approach has advantages since it utilizes large, accumulated data and depends less on the user's experience. Therefore, data-driven machine learning models are utilized for the estimation of the initial parameters of aircraft conceptual design.

Several machine learning models, k-nearest neighbors (kNN), variational autoencoder (VAE), and random forest (RF) are applied. These models are capable of handling incomplete and heterogeneous data. This is a huge advantage because most of the existing aircraft data are incomplete and composed of various data types. An incomplete dataset can be entered so that the model can directly learn from the data. The actual data of the aircrafts that served in World War II is gathered and utilized in this study. kNN, VAE, and RF have distinct learning tactics, so this study set the universal criteria to compare the performances of the three models. Furthermore, unlike other studies that require a complete dataset for training and validation, this study constructed the repeated imputation procedure and successfully assessed both models. As preliminary research, this study presented the validity of applying a data-driven approach to the aircraft initial sizing problem.

Keywords: aircraft conceptual design, initial parameter estimation, k-nearest neighbors, variational autoencoder, random forest

1. Introduction

The aircraft design process consists of three major phases: conceptual design, preliminary design, and detail design [1]. The conceptual design phase is ahead of the overall design process, so the results should be provided fast so that the modification can be reflected in the following stages. The conceptual design process requires the design requirements and then explores possible sets of specifications such as configuration, weight, and performance. Thus, defining an adequate design space is an essential process for attaining a desirable design result.

Initial design parameter estimation is carried out concurrently with the conceptual sketch of the initial layout. The optimization process is followed based on this initialization. Therefore, during the aircraft designing process, selecting the initial parameters preceding the conceptual design phase has been an essential step in establishing the overall outline of the aircraft to be developed.

Existing methods have used empirical formulas with assumptions or adopted empirical parameters to set the initial parameters. These conventional methods have limitations in that only an insufficient small subset of the design space is explored because they reflect very few reference cases. This leads to an ineffective design because the user might start exploring at the wrong point and end up calculating at a local minimum. While setting reference cases, the design process and result unavoidably become dependent on the engineer's design experiences and background knowledge. For example, the accuracy of the linear regression highly depends on the similarities between the input data and the target configuration. Moreover, when an empirical formula is established by combining variables, the calculating process is inefficient because each formula has a different objective function. Moore et al. [2] developed a design tool for aircraft conceptual design studies based on the optimization of the objective functions. Similarly, Ardema et al. [3] estimated relations between the load-bearing fuselage weight, wing weight, and other related weights by deriving the empirical formulas. However, the derivation of numerous empirical equations regarding diverse combinations of variables in serial order was inevitable. In such a problematic situation, the limitations of the existing method can be overcome if a data-driven method is applied.

The data-driven approach has various advantages. First of all, it can utilize a large amount of accumulated data when designing a new case. This means that the design result no longer adopts a single reference but reflects the overall tendency of cumulated data. Thus, the process gets less dependent on user knowledge and experience, facilitating efficient design space exploration. Also, models with various inputs and outputs are implementable since the user can make flexible changes to the model.

Despite these advantages, data-driven methods have not been utilized in the initial parameter setting of aircraft conceptual design. The existing data-driven parameter estimation methods require or assumed as their dataset are complete or homogeneous for training and evaluation. [4] Peyada and Ghosh [5] successfully generated the flight data by solving the equations of motions directly instead of gathering actual experimental data, yet the incomplete dataset could not be entered directly into the model.

In the real-world application, most of the aircraft databases are incomplete. This is because the aircraft specification details are usually incomplete to maintain security due to the nature of the aviation industry. Also, the types of available data vary from aircraft to aircraft because the aircraft design is extremely complex. For example, the aircraft design parameter dataset gathered from [6] is composed of 85 categories in total. For these reasons, most of the existing aircraft databases are incomplete and heterogeneous, and thus the conceptual design process could not be fused with data-driven methods. An example of the existing aircraft data is demonstrated in Table 1. It is a brief overview of the dataset of aircraft that served in World War II.

Table 1 – Brief overview of the dataset of aircraft of World War II

	Real-valued				**Categorical**	**Countable**
	Total power [h.p.]	Span [m]	Wing area [m^2]	Empty weight [kg]	Landing gear type	Number of engines
1	65	7.22	?	285	fixed	1
2	520	17.02	43.85	2791	fixed	?
3	?	13.1	34	?	retractable	2
4	?	31.7	1184	17360	?	4
5	2400	12.5	28.06	?	retractable	1

Data-driven machine learning models can be a solution for the limitations of the existing data-driven methodologies. The presenting research overcame the limitations mentioned above by utilizing those machine learning models to estimate the initial parameters of aircraft conceptual design. In this study, the k-nearest neighbors algorithm (kNN), variational autoencoder (VAE), and random forest (RF) are applied among numerous machine learning models, to estimate the initial parameters of the aircraft conceptual design.

The three models – kNN, VAE, and RF – have remarkable advantages. First of all, they can deal with data with inconsistent types. Thus, they can also handle high-dimensional multivariate data. Also, they can be trained with incomplete data and consequently fill in the missing attributes. These valuable properties enable the design parameter estimation by applying data-driven methods to real-world aircraft data.

In this research, we propose preliminary research presenting the validity of applying a data-driven approach to the aircraft initial sizing problem. To verify whether data-driven machine learning methods are appropriate for handling the existing aircraft data, missing data estimation experiments are conducted. Feasible combinations of input variables are suggested as a result. Furthermore, the performances of kNN, VAE, and RF models are compared. The goal of this study is to enable the utilization of aircraft databases by imputation techniques when selecting the initial parameters of the conceptual aircraft design process.

2. Methodology

2.1 Overview of the comprehensive aircraft design process

The aircraft design process consists of three main phases: conceptual design, preliminary design, and detail design. Setting and adjusting the desired requirements should precede the conceptual design stage. The performance or the specifications such as weights should comply with the design requirements, and the cost should be compromised along with these results.

The user explores the possible geometry and following performance of an aircraft that is to be designed in the conceptual design phase. In the preliminary design, major decisions on the aircraft are done. The details of the aircraft design are realized during this step. Then the detail design phase begins by analyzing and scrutinizing the separate elements. The user continuously monitors if the design still meets the original requirements. The performances of the actual aircraft are tested and modified as a final step. The fabrication of the aircraft follows after these design procedures are done.

The optimization techniques are used in in the conceptual design phase. For the accuracy of the resultant, the calculation loads are very big and therefore the optimization processes are usually costly. One of the best ways to reduce the computational load is to start the optimization with appropriate design space. Furthermore, starting the aircraft parameter estimation with suitable experimental design can lead to more accurate estimation. [7]

In this sense, this research provides a feasible design space for the variables in the conceptual design. This study concentrates on handling incomplete and heterogeneous aircraft data by using data-driven machine learning methods. The overall flow of aircraft designing phases is demonstrated in Figure 1.

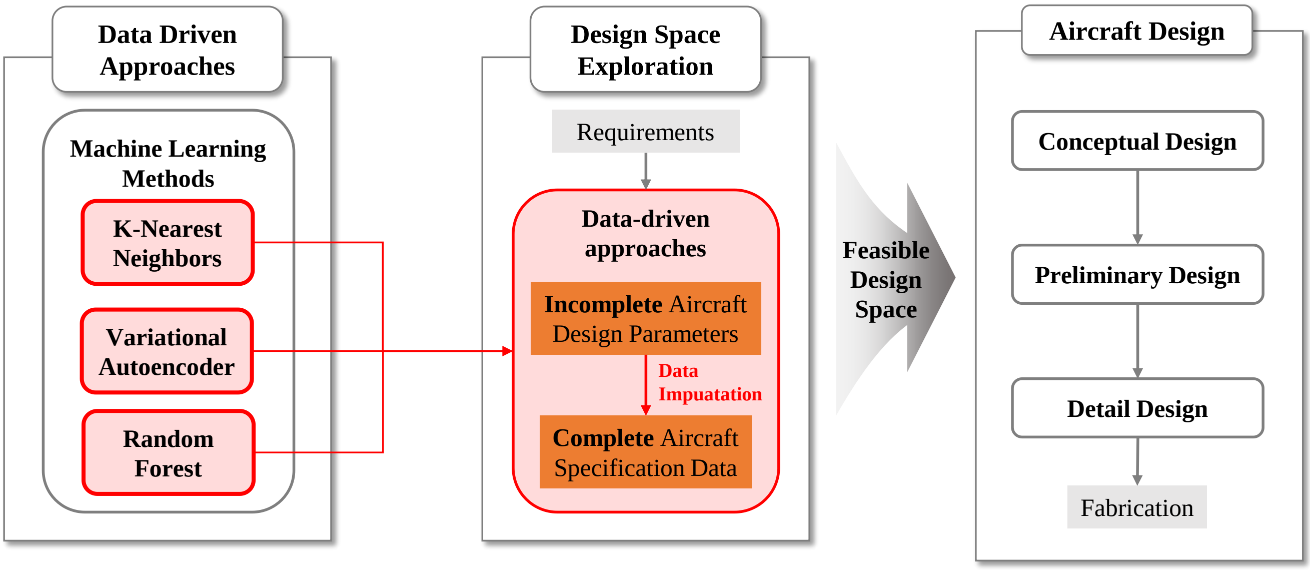


Figure 1 – Overall flow of the whole framework

2.2 Incomplete data processing

There are two ways of processing incomplete data. First of all, the user can exclude incomplete attributes from the dataset. The user can either drop the missing variables or exclude all the incomplete cases. Both of these options are mostly disadvantageous since the data can be biased or the size of the dataset decreases [8, 9]. Second, the data imputation. There are many ways of filling in the missing attributes: mean, median, mode, random sample imputation, linear interpolation, linear regression, and other sophisticated imputation methods such as using machine learning models. Although simple statistical imputation methods such as mean, median, or mode imputation are better than deleting methods, they usually have low precision because they merely fill in a uniform value. These methods are prone to be affected by outliers. Linear interpolation and linear regression are also vulnerable to outliers, and their performance hugely depends on the distribution of the given data. In this regard, users need a more sophisticated and solid method for missing data imputation. Among various methods, data imputations using machine learning models are implemented.

2.3 K-Nearest Neighbors Algorithm

The k-nearest neighbors (kNN) algorithm is a fundamental and simple data-driven method that enables fast implementation. The algorithm is firstly suggested by Evelyn Fix and Joseph Hodges [10], and applied as a predicting tool [11], classifier [12, 13], numerical regressor [14, 15, 16], imputer [17, 18, 19]. It utilizes the local information of k-nearest samples by vectorizing all samples and calculating distances between every vector. The distance is calculated by using the distance metrics such as Minkowski distance, Euclidean distance, and so on.

The precision depends on the number of the nearest neighbors setting. The appropriate k depends on the characteristic of each data. Generally, models with larger k are less affected by the outliers or noise, but the distinction between different classes becomes obscure. On the other hand, models with smaller k are more sensitive to the data distribution because it is more likely to get influenced by the outliers.

kNN can handle heterogeneous data by taking different strategies for each data type. In this study, kNN regression and kNN classification are used. The output of kNN regression is a real value. To estimate the missing attribute, values of the k nearest neighbors are averaged. The process is iterated until it meets the criteria. In this study, the model stops its iteration when Equation 1 is satisfied where X_n is a resultant value X at iteration n. The kNN regression process is depicted in Figure 2.

$$max\ (|X_n - X_{n-1}|)\ /(|X_{max}|) < tolerence \tag{1}$$

The output of kNN classification is a class label. The distance between each case is the same every time, so the kNN classification iteration is not needed. The kNN classification process is demonstrated in Figure 3. To impute regression and classification data jointly, an integrated kNN sequential process is set. An incomplete regression data is imputed firstly, and the resultant complete regression data is coupled with incomplete classification data for the following kNN classification. Therefore, the performance of kNN regression affects the results of kNN classification. Missing attributes are considered as the value of 0. The flowchart of the integrated kNN sequential process is shown in Figure 4.

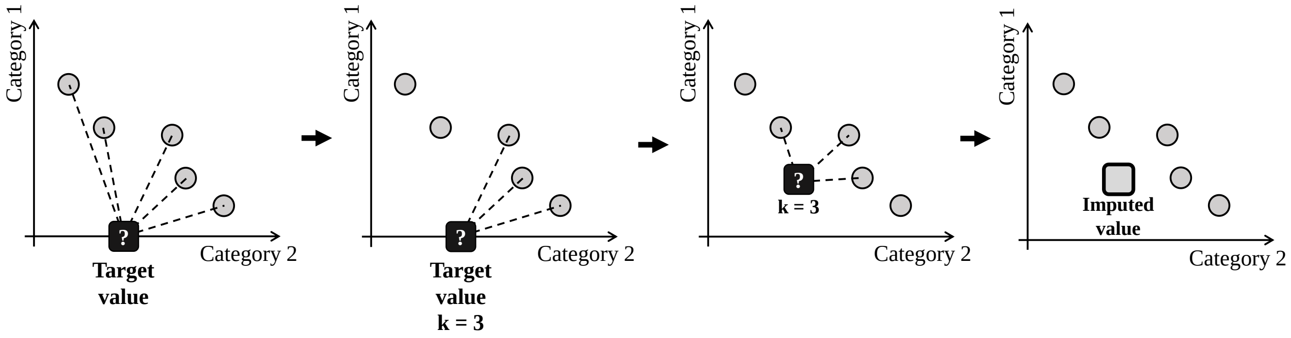


Figure 2 – kNN regression process

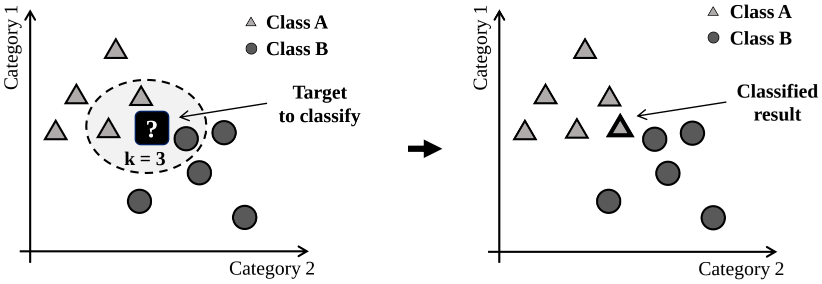


Figure 3 – kNN classification process

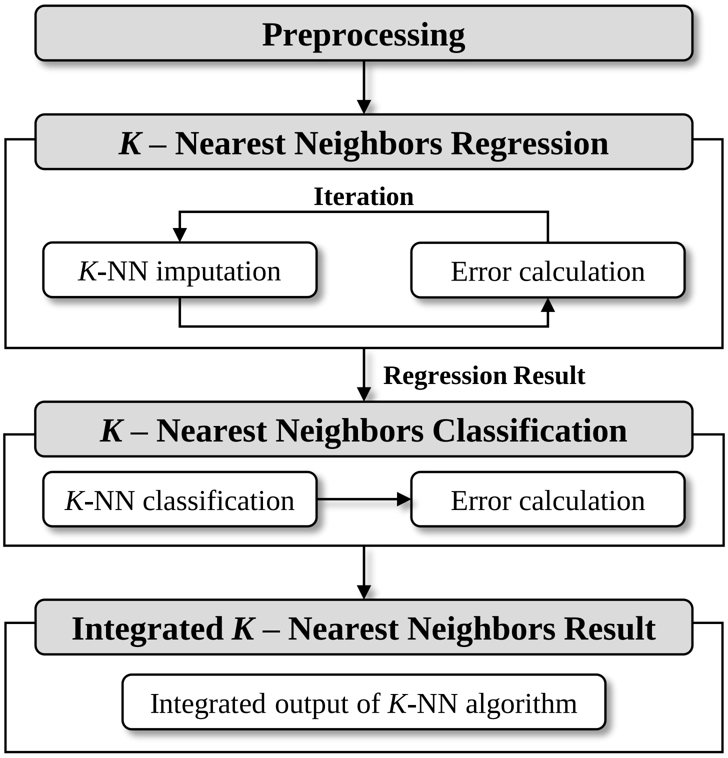

Preprocessing
K – Nearest Neighbors Regression
Iteration
K-NN imputation
Error calculation
Regression Result
K – Nearest Neighbors Classification
K-NN classification
Error calculation
Integrated K – Nearest Neighbors Result
Integrated output of K-NN algorithm

Figure 4 – Flowchart of the integrated kNN sequential process

2.4 Variational Autoencoder

2.4.1. Generative Model

Generative models are capable of generating realistic new data. A generative model learns and generates new data that is similar to the true data. Generative adversarial network (GAN) [20] and variational autoencoder (VAE) [21] are the representatives of deep generative models. GAN is composed of the generator and discriminator. The generator generates new fake data, and the discriminator tries to distinguish the real data. The situation can be interpreted as a two-player minimax game. GAN uses backpropagation to earn the gradient, so it can generate samples without using the Markov chain, and no inference is needed during the training. However, GANs have some disadvantages, including the vanishing gradients, oscillation during the optimization process, and mode collapse problems. Furthermore, a newly proposed GAN application, GAIN [22] can complete incomplete input data by using the generator. This model also has a shortage in that it can only handle limited types of data such as binary data. For these reasons, VAE is utilized in this study.

2.4.2. Variational Autoencoder

Variational Autoencoder (VAE) is one of the representative deep generative models. VAE reads and emulates the underlying distributions of the original data. VAE consists of the encoder and decoder. The encoder is a recognition network that extracts the features out of the input x and generates latent variables z. Among these encoded results, a set of mean and variance is randomly sampled to form a latent space. Then, the decoder generates the output according to the given latent variables. The operating flow of VAE is depicted in Figure 5.

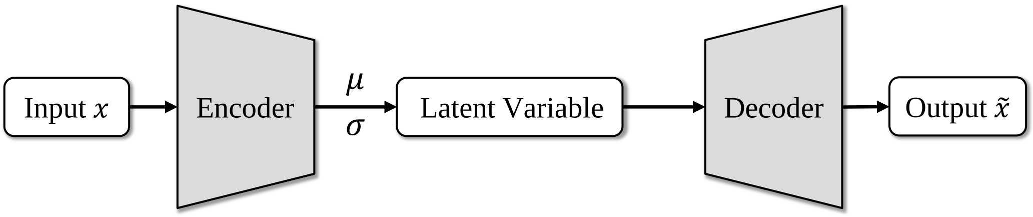


Figure 5 - Flowchart of VAE

Let us consider that there is the dataset $\mathrm{X} = \{x_i\}_{i=1}^{N}$ that consists of N i.i.d. (independent and identically distributed) samples of input variable x and an unobserved random variable z that produces x. The variable z is also known as the latent variable. The posterior distribution can be denoted as Equation 2, and it is intractable, where θ and ϕ are the generative model parameters and recognition model parameters, respectively.

$$p_\theta(z|x) = \int_z \frac{p_\theta(X|Z)p_\theta(z)}{p_\theta(x)} dz \tag{2}$$

Variational inference is used since the posterior is intractable. Thus, variational approximation $q_\phi(z|x)$ is defined to be considered as an approximation of the posterior $p_\theta(z|x)$. $q_\phi(z|x)$ can also be denoted as a recognition model, encoder.

The likelihood function represents how much a particular population is likely to produce an observed sample. It is a form of a joint density function. The marginal likelihood can be defined as Equation 3.

$$log\ p_\theta(x_i) = D_{KL}\left(q_\phi(z|x_i) \parallel p_\theta(z|x_i)\right) + \mathcal{L}(\theta, \phi; x_i) \tag{3}$$

$D_{KL}\left(q_\phi(z|x_i) \parallel p_\theta(z|x_i)\right)$ is the Kullback-Leibler (KL) divergence of the approximation $q_\phi(z|x)$ from the true posterior $p_\theta(z|x)$. KL divergence measures how two probability distributions p_θ and q_ϕ

are different from each other. $\mathcal{L}(\theta, \phi; x_i)$ is known as the variational lower bound on the marginal likelihood. KL divergence is non-negative and is equal to zero if two distributions are identical, and thus Equation 4 follows.

$$log\, p_\theta(x_i) \geq E_{q_\phi(z|x_i)}[log\, p_\theta(x_i|z)] - D_{KL}\left(q_\phi(z|x_i) \parallel p(z)\right) \qquad (4)$$

Equation 4 is the evidence lower bound (ELBO) of the marginal likelihood. The first term on the right-hand side of the inequality is the reconstruction error. The model should have objectives of minimizing the KL divergence and maximizing the likelihood function and ELBO.

2.4.3. Heterogeneous-Incomplete Variational Autoencoder

Although VAE is an elaborate model with an outstanding performance, particular VAEs are not capable of dealing with incomplete or heterogeneous (mixture of discrete and continuous) data. However, handling incomplete data is a critical task in this study. Thus, the Heterogeneous-Incomplete Variational Autoencoder (HI-VAE) [23] is chosen among the application of VAEs for this study. The flowchart of HI-VAE is demonstrated in Figure 6.

HI-VAE separates observed and missing attributes, and pre- and post-processes each data type separately. Pre-processed attributes enter the encoder and decoder together. Missing attributes are initially considered as a value of zero and have no contribution to the training. The decoder is factorized as Equation 5.

$$p(x_n, z_n) = p(z_n) \prod_d p(x_{nd}|z_n) \qquad (5)$$

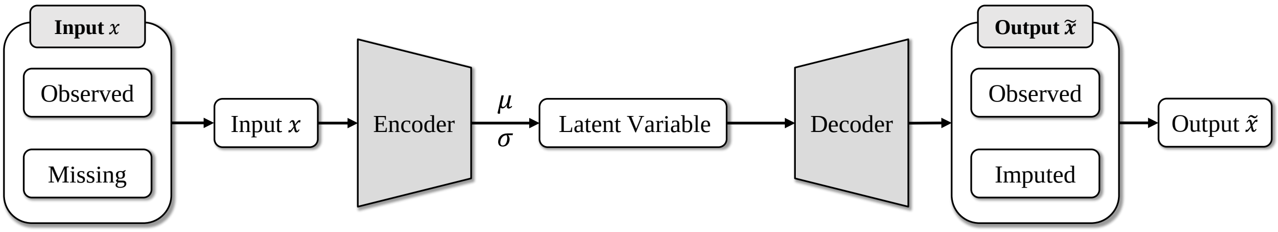


Figure 6 - Flowchart of HI-VAE

Table 2 – Likelihood models for each data type

Real-valued data	Likelihood model	Gaussian likelihood model
	Relation	$p(x_{nd}|\gamma_{nd}) = \mathcal{N}(x_{nd}|\mu_d(z_n), \sigma_d^2(z_n))$ $\gamma_{nd} = \{\mu_d(z_n), \sigma_d^2(z_n)\}$
Positive real-valued data	Likelihood model	Log-normal likelihood model
	Relation	$p(x_{nd}|\gamma_{nd}) = \log\mathcal{N}(x_{nd}|\mu_d(z_n), \sigma_d^2(z_n))$ $\gamma_{nd} = \{\mu_d(z_n), \sigma_d^2(z_n)\}$
Count data	Likelihood model	Poisson likelihood model
	Relation	$p(x_{nd}|\gamma_{nd}) = Poiss(x_{nd}|\lambda_d(z_n))$ $\gamma_{nd} = \lambda_d(z_n)$
Categorical data	Likelihood model	Multinomial logit model
	Relation	$p(x_{nd} = r|\gamma_{nd}) = \frac{\exp(-h_{dr}(z_n))}{\sum_{q=1}^{R}\exp(-h_{dq}(z_n))}$ $h_{d0}(z_n) = 0$
Ordinal data	Likelihood model	Ordinal logit model
	Relation	$p(x_{nd} = r|\gamma_{nd}) = p(x_{nd} \leq r|\gamma_{nd}) - p(x_{nd} \leq r-1|\gamma_{nd})$ $p(x_{nd} \leq r|\gamma_{nd}) = \frac{1}{1+\exp(-(\theta_r(z_n) - h_d(z_n)))}$
Definition of variables	d=dimension of the vector n=the numbering of the objects γ=likelihood parameter μ=mean σ^2=variance	

HI-VAE applies different likelihood models for each data type. Likelihood models are demonstrated in Table 2. HI-VAE divides data into numerical and nominal variables. There are three types of numerical variables: real-valued data, positive real-valued data, and discrete countable data. Countable data is a positive integer. Also, there are two types of nominal variables: categorical data and ordinal data. For example, categorical data can be a set of 'yellow, red, and blue' attributes and ordinal data can take values with orders such as 'low wing, mid wing, and high wing'.

The model processes all data in a form of numbers. Each attribute of categorical and ordinal data is therefore given a distinct number. For example, the low wing is considered as a value of 1, mid wing as 2, and high as 3. Nominal variables are processed based on different log-likelihood functions, thus the numbers given to those variables are distinct from that of the numerical variables.

2.5 Random Forest

2.5.1 Decision Tree, Random Forest, and MissForest

A decision tree (DT) is one of the most widely used non-parametric supervised learning algorithms. Decision trees are typically used for divide-and-conquer classification and regression problems. The algorithm divides the given input data from the root node according to certain criteria until it terminates the training at the leaf nodes. There are various versions of decision tree algorithms that enable effective decision-making. Among them, Iterative Dichotomiser 3 (ID3) [24], Classification 4.5 (C4.5) [25], Classification and Regression Trees (CART) [26] algorithms are extensively used. Decision trees are capable of serving the following tasks: variable selection [27, 28, 29], correlation analysis between variables, prediction [30, 31], incomplete data imputation [32], data manipulation, and so on.

Both the regression trees and classification trees are built based on recursive partitioning. Recursive partitioning is a process that splits the feature space into subsets that contains attributes with similar traits. The procedure repeats until it reaches the stop condition. The user can effectively understand the decision-making recursive partitioning process because the decision tree model is an interpretable white-box model. The result of a decision tree is simple, straightforward, and easy to visualize. The model can handle both numerical and categorical data and does not require any assumptions regarding the data distribution since it is a non-parametric model. On the other hand, there are some disadvantages of a single decision tree model: it can only deal with one type of data at a time, it is unstable that the result can be biased by the outliers, and the result is inaccurate especially when the relationship between the input variables is complex. In this sense, a decision tree model is considered a weak learner.

To overcome these limitations, Breiman [33] developed the random forest (RF) by reflecting multiple results of classification and regression trees (CART). The random forest training is done by a bootstrap aggregation (bagging) process; the result of each decision tree is integrated to make the resultant model robust. Bootstrap aggregation, so-called bagging, is based on random sampling with the replacement of the training dataset. Multiple decision trees are trained in parallel by the random sampled subsets.

The use of multiple trees to form a random forest model can lead to inefficient and complicated computation. It also causes a lack of interpretability unlike a model composed of a single decision tree. Despite several disadvantages, there are useful advantages of random forests.

First of all, bagging makes the random forest model more robust to the overfitting issues. The variance of the prediction model decreases by utilizing a number of subsets, the model becomes capable of dealing with overfitting. Also, a random forest model can deal with continuous numerical data and discrete classification data simultaneously. This is very advantageous since real-world data are composed of various types of data in many cases.

This study used the MissForest model, which is a specified version of the RF model. Missforest (MF) [34] is based on a random forest and is suitable for incomplete data imputation. It trains the model with the observed values first and iteratively imputes the missing values. The MissForest imputation flowchart and process demonstration are shown in Figures 7 and 8, respectively. The columns are rearranged according to the missing rates (Figure 8(b)), the mean imputed for the initial guess (Figure 8(c)), and the rest of the data are used for the target attribute estimation (Figure 8(d)). The whole process is repeated until it reaches a stopping criterion. Stopping criterion is met when the difference between the results of previous and current time steps increases for the first time. The difference between the continuous numerical values, Δ_{num}, and discrete categorical data, Δ_{cat}, are demonstrated in Equation 6 and 7, respectively, where X_t denotes the imputation result at a time step t, and $N_{miss,cat}$ denotes the number of missing attributes in the discrete categorical data.

$$\Delta_{num} = \frac{\sum_{i \in C} (X_t - X_{t-1})^2}{\sum_{i \in C} (X_t)^2} \tag{6}$$

$$\Delta_{cat} = \frac{\sum_{j \in D} \sum_{i=1}^{n} I_{X_t \neq X_{t-1}}}{N_{miss,cat}} \tag{7}$$

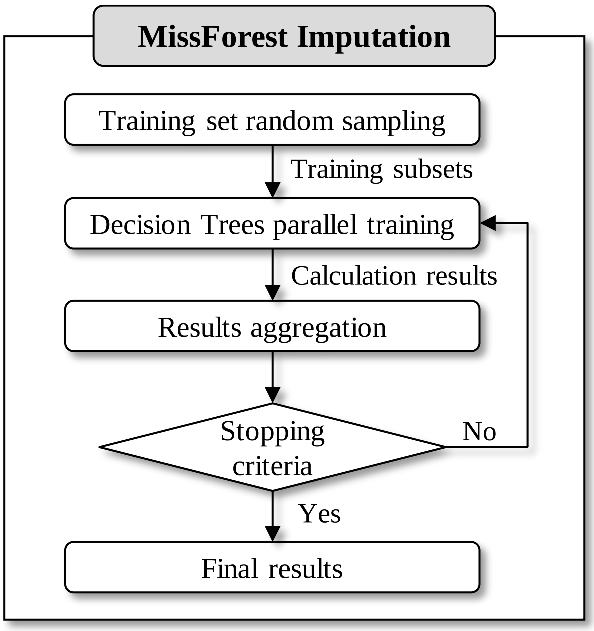


Figure 7 – MissForest data imputation flowchart

No.	Span [m]	Wing Area [m^2]	Wing Position
1	17	?	Low
2	27	112	?
3	22	43	Mid
4	16	?	Shoulder
5	?	105	High
6	7	?	High

(a)

No.	Span [m]	Wing Position	Wing Area [m^2]
1	17	Low	?
2	27	?	112
3	22	Mid	43
4	16	Shoulder	?
5	?	High	105
6	7	High	?

(b)

No.	Span [m]	Wing Position	Wing Area [m^2]
1	17	Low	*87*
2	27	*High*	112
3	22	Mid	43
4	16	Shoulder	*87*
5	?	High	105
6	7	High	*87*

(c)

No.	Span [m]	Wing Position	Wing Area [m^2]
1	17	Low	87
2	27	High	112
3	22	Mid	43
4	16	Shoulder	87
5	***24***	High	105
6	7	High	87

(d)

Figure 8 – MissForest data imputation process

3. Experimental Setup

3.1 Dataset

In this study, Jane’s World War II (WWII) dataset with 228 aircraft cases is gathered from the book <Jane’s Fighting Aircraft of World War II>. The book contains assorted information about the aircraft that served in World War II. Aircraft cases included in this book had similar purposes of serving in the war. Therefore, there was no need to classify the data as their distinct purposes.

We gathered usable aircraft cases with sufficient specifications. Then one countable, three categorical, and nine positive-valued categories are selected among 85 available categories because they have missing rates no higher than 30 percent. Types and missing rates of these 13 categories are summarized in Figure 9 and Table 3.

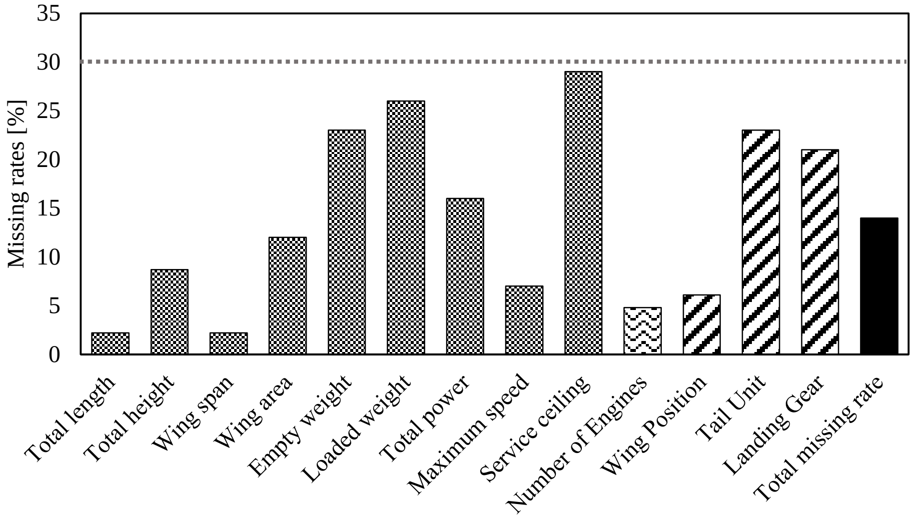


Figure 9 - Missing rates of each category in the WWII dataset

Table 3 - Summary of the WWII data

Data types	Classes	Missing rates [%]
Real-valued	Total length	2.2
	Total height	8.7
	Wing span	2.2
	Wing area	12
	Empty weight	23
	Loaded weight	26
	Total power	16
	Maximum speed	7.0
	Service ceiling	29
Countable	Number of Engines	4.8
	Wing Position	6.1
	Tail Unit	23
	Landing Gear	21
Categorical	Total missing rate	14

3.2 Training Conditions

Hyperparameters such as epochs, latent dimension, and batch size are tuned elaborately based on the training objectives.

The number of nearest neighbors in kNN experiments is chosen based on the heuristic experiments. The value of k is set as 5. For kNN experiments, data are normalized before the imputation to avoid bias among different data types with different scales. All data are normalized on a scale between -1 to 1 since the unknown attributes are considered as 0.

If a model is trained excessively with a certain dataset, the model can be overly fit only to that dataset. Under this circumstance, the model performance can be poor when new data is entered. This is known as overfitting, and early stopping is a solution to avoid overfitting. When the early stopping is applied, it allows the model to update the training parameter and continue to iterate the process until it begins to over-fit. The early stopping criterion is applied for kNN. The maximum iteration of kNN is 300 and the number of epochs of HI-VAE is set as 500. For VAE experiments, data is entered with its original scale since the model contains a batch normalization layer.

VAE uses the LeakyReLu as an activation function and Mean Squared Error as a loss function in this study. The latent dimensions of VAE are all set as 100.

The hyperparameters of the RF are selected by using Bayesian optimization. The number of trees in the forest is set as 138, the minimum number of samples required at a leaf node is set as 2, and the minimum number of samples required to split an internal node is set as 2. The Missforest model can directly deal with the original scale data.

3.3 Standard Error Criteria

kNN, VAE, and RF have distinct imputation procedures and tactics. The objective function of kNN is based on the difference between the actual values of the overall imputed data. On the other hand, VAE updates its weights based on the training, test, and validation loss, and RF decides whether to split according to the numerical difference between time steps. Hence, there is a need to establish standard error criteria to compare the two models’ performance properly. From the user’s point of view, the information of the calculated output is one of the most accessible data.

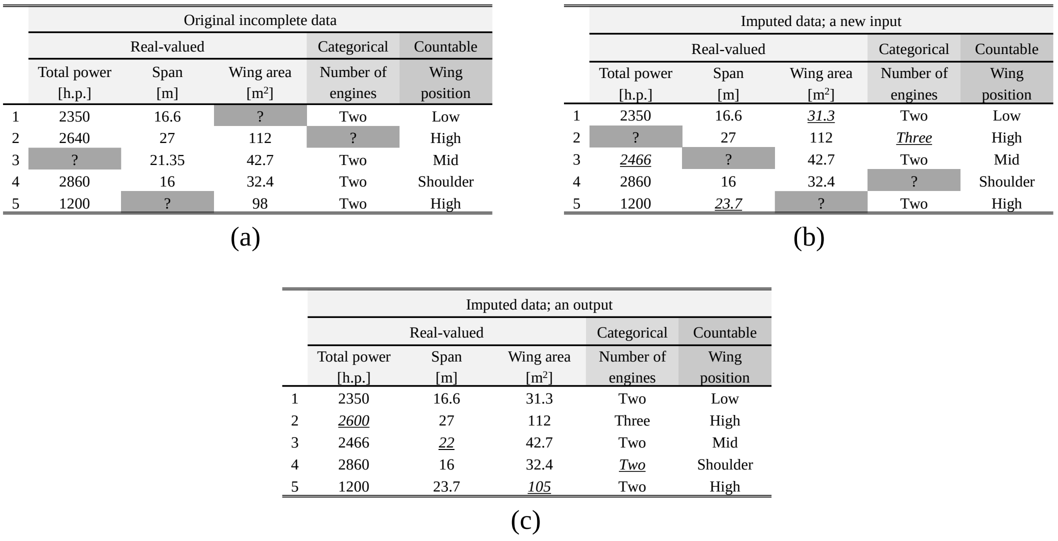
Therefore, two error indicators monitoring the difference between the true and predicted values are defined: Mean Average Percentage Error (MAPE) and Proportion of Falsely Classified (PFC) entries. They indicate regression and classification errors, respectively. The ways of calculating MAPE and PFC are demonstrated in Eq 6 and Eq 7.

$$E_{MAPE} = avg\left(\left|\frac{true - prediction}{true} \times 100\right|\right)\ (\%) \tag{6}$$

$$E_{PFC} = \left(\frac{\#\ of\ wrong\ predictions}{\#\ of\ total\ missing\ cases}\right) \times 100\ (\%) \tag{7}$$

3.4 Validation

Existing studies regarding the missing data imputation validate their studies by using complete datasets. However, the aircraft dataset used in this study does not have a full version of itself. Thus, there is a need of defining a distinct way of validation to assess the result of this study. We suggest a repeated imputation validation procedure, which repeats the imputation procedure twice. This procedure considers the firstly imputed dataset as a true value. Then, the values other than the initially missing attributes are removed according to the original missing rates of each category. This way, MAPE and PFC errors can be calculated. The repeated imputation procedure is demonstrated in Figure 10. Based on the repeated imputation validation procedure, two versions of the WWII dataset are imputed. The first one is a complete dataset that is organized by removing all the incomplete cases. The other one is the original WWII dataset containing 228 cases. Once the imputation procedure is done, the completed full dataset is given as an output. Datasets are modified to have 5, 10, 15 percent of missing rates by removing some random attributes other than the original missing ones from the imputed full datasets.



	Original incomplete data				
	Real-valued			Categorical	Countable
	Total power [h.p.]	Span [m]	Wing area [m²]	Number of engines	Wing position
1	2350	16.6	?	Two	Low
2	2640	27	112	?	High
3	?	21.35	42.7	Two	Mid
4	2860	16	32.4	Two	Shoulder
5	1200	?	98	Two	High

(a)

	Imputed data; a new input				
	Real-valued			Categorical	Countable
	Total power [h.p.]	Span [m]	Wing area [m²]	Number of engines	Wing position
1	2350	16.6	*31.3*	Two	Low
2	?	27	112	*Three*	High
3	*2466*	?	42.7	Two	Mid
4	2860	16	32.4	?	Shoulder
5	1200	*23.7*	?	Two	High

(b)

	Imputed data; an output				
	Real-valued			Categorical	Countable
	Total power [h.p.]	Span [m]	Wing area [m²]	Number of engines	Wing position
1	2350	16.6	31.3	Two	Low
2	*2600*	27	112	Three	High
3	2466	*22*	42.7	Two	Mid
4	2860	16	32.4	*Two*	Shoulder
5	1200	23.7	*105*	Two	High

(c)

Figure 10 – Repeated imputation procedure

4. Result and Discussions

4.1 Variational missing rate

To verify the repeated imputation procedure, the same imputation experiments are conducted for the new versions of two WWII datasets. One dataset is only composed of 80 complete cases by excluding all the incomplete cases. This way, we can obtain the complete real aircraft data. The other one is the original incomplete dataset that consists of 228 cases.

Both datasets are imputed once to form complete versions of themselves. Then, we eliminated some attributes from which the datasets have missing rates of 5 %, 10 %, and 15 %. The originally missing positions are not deleted again. The error would increase if the missing rate rose because there are fewer references that the calculation process can utilize. In this sense, if the error increment trend is similar, then it can support that the repeated imputation procedure is valid.

As a result, similar trends of MAPE error increments between the dataset composed of 80 complete cases (Figure 11(a)) and the dataset composed of 228 incomplete cases (Figure 11(b)) are observed. In both cases, RF had the greatest errors and VAE had the least errors. This tendency can be interpreted as one of the proofs that this validation tactic is reasonable.

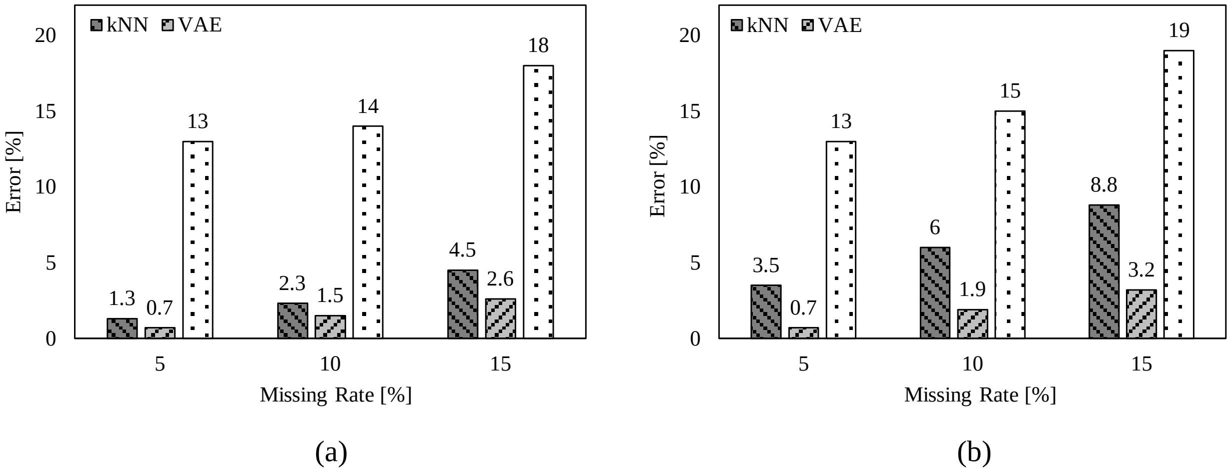


Figure 11 – MAPE errors using (a) 80 complete and (b) 228 incomplete cases

4.2 Original missing rate

The actual MAPE and PFC errors of the WWII original dataset with an original missing rate are indicated in Table 4 and plotted in Figure 12. Both the MAPE and PFC errors of the VAE were about half of those of the kNN. Both kNN and VAE had better performance on regression tasks than the classification tasks. However, RF had the opposite tendency compared to the other models. RF had the worst regression estimation while having the best classification performance. This can be interpreted as the RF regression tactic is not appropriate for the datasets that are similar to Jane's WWII dataset used in this study. RF regression partitions the input and then averages the values in each subgroup. However, Jane's WWII dataset only contains 228 cases with similar tendencies since the dataset is composed of aircrafts that served similar purposes during the Second World War. Thus, the numerical values have biased tendencies. Partitioning the biased regression data into small subgroups and averaging their information can cause a wrong estimation. The characteristics of the dataset and the regression tactic have led the RF model regression inaccurate.

A specific example of M-14 from The Miles is demonstrated in Figure 16. The service ceiling information is imputed in the first place and regarded as true data since it was unavailable in the book. To calculate the model error, information about the empty weight is removed and then the imputation process is repeated. In this sense, the MAPE error is calculated by comparing the original and the estimated value. It is clear that the VAE performance excels that of the kNN and RF in regression problems.

The VAE training is monitored by the log-likelihood function. In this experiment, log-likelihood functions for the positive-real value, countable, and categorical types are used. The maximization of the likelihood is observed, so it can be concluded that the optimization process is successfully implemented. The log-likelihood functions regarding the epochs are demonstrated in Figure 13. Values are averaged according to their data types.

From the user's perspective, the VAE and RF models are much easier to use than the kNN model because it can handle various types of data at the same time. The user should complete the kNN regression and then continuously add single categories listwise. The calculation process becomes ineffective. Figure 14 depicts the situation where the kNN classification of the countable data – the number of engines – is done after the kNN regression. The user still has to go through the remaining classification procedures. After several classification procedures, a comprehensive result is obtained. The part of the comprehensive result is depicted in Figure 15. On the other hand, VAE and RF can output a complete result at once. In other words, they do not need to undergo the process illustrated in Figure 14.

Table 4 – MAPE and PFC errors with the original missing rate

Model	E_{MAPE} [%]	E_{PFC} [%]
K-NN	6.1	13
VAE	2.8	13
RF	17	3.5

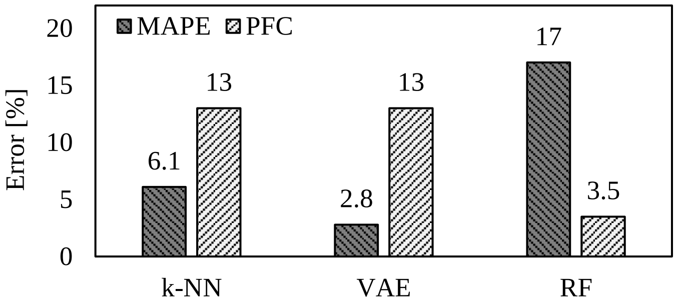
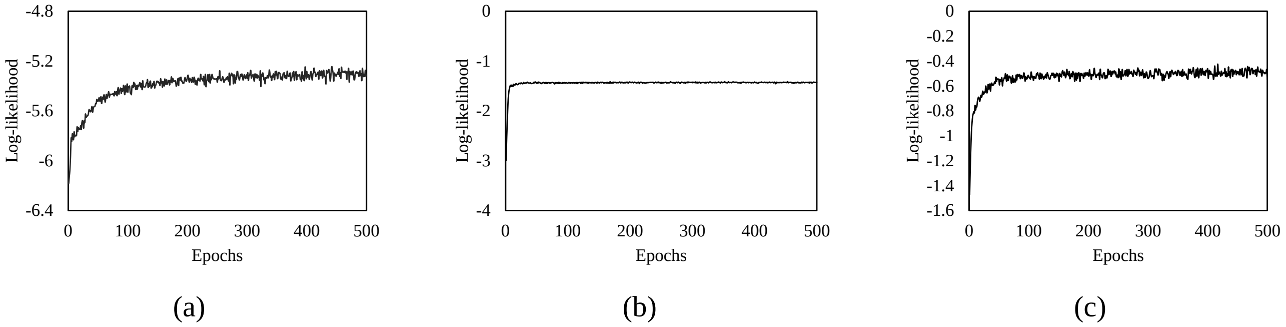


Figure 12 – MAPE and PFC errors using 228 incomplete cases



(a) (b) (c)

Figure 13 – Log-likelihood of (a) positive real-valued, (b) countable, and (c) categorical data

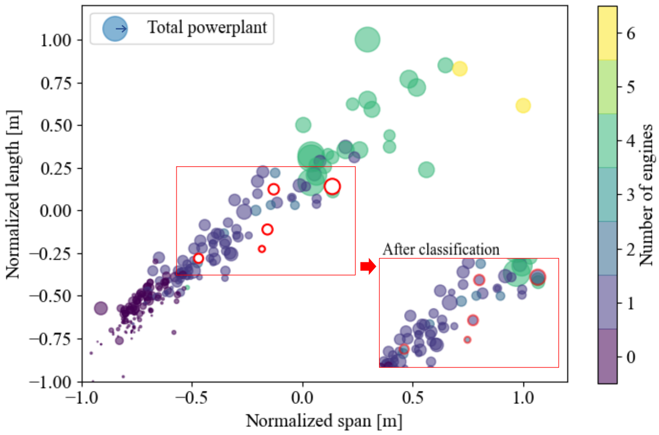


Figure 14 – Intermediate result of the kNN sequential imputation process

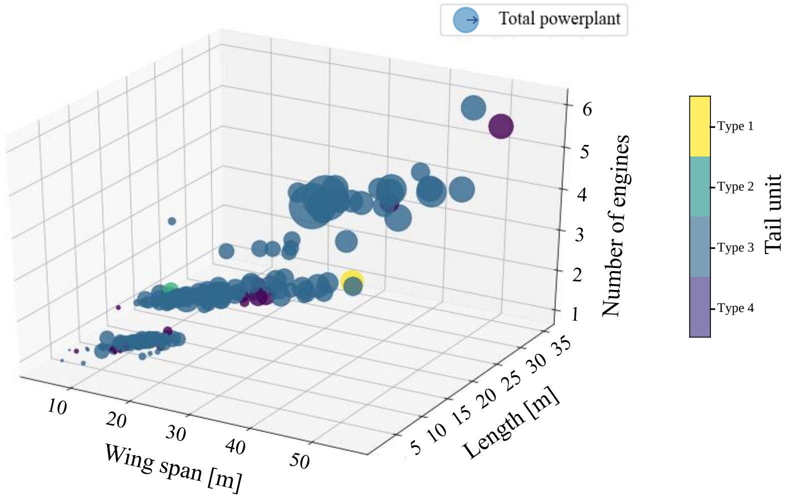
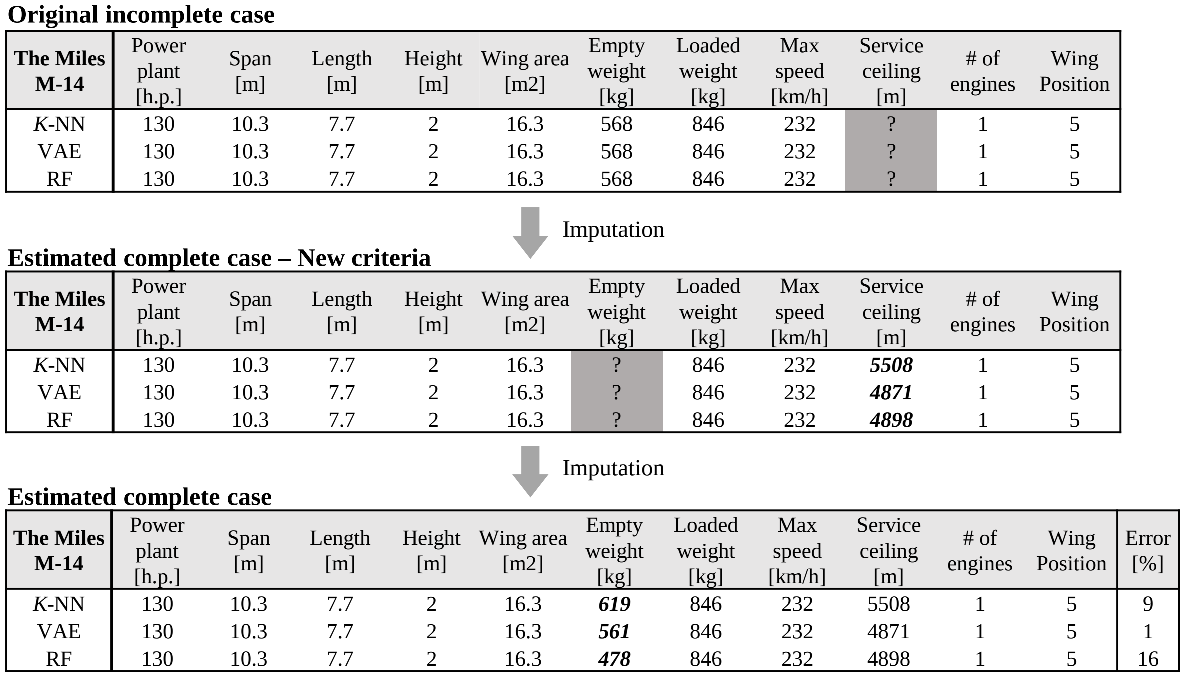


Figure 15 – Partial demonstration of imputation result



Original incomplete case

The Miles M-14	Power plant [h.p.]	Span [m]	Length [m]	Height [m]	Wing area [m2]	Empty weight [kg]	Loaded weight [kg]	Max speed [km/h]	Service ceiling [m]	# of engines	Wing Position
K-NN	130	10.3	7.7	2	16.3	568	846	232	?	1	5
VAE	130	10.3	7.7	2	16.3	568	846	232	?	1	5
RF	130	10.3	7.7	2	16.3	568	846	232	?	1	5

Imputation

Estimated complete case – New criteria

The Miles M-14	Power plant [h.p.]	Span [m]	Length [m]	Height [m]	Wing area [m2]	Empty weight [kg]	Loaded weight [kg]	Max speed [km/h]	Service ceiling [m]	# of engines	Wing Position
K-NN	130	10.3	7.7	2	16.3	?	846	232	***5508***	1	5
VAE	130	10.3	7.7	2	16.3	?	846	232	***4871***	1	5
RF	130	10.3	7.7	2	16.3	?	846	232	***4898***	1	5

Estimated complete case

The Miles M-14	Power plant [h.p.]	Span [m]	Length [m]	Height [m]	Wing area [m2]	Empty weight [kg]	Loaded weight [kg]	Max speed [km/h]	Service ceiling [m]	# of engines	Wing Position	Error [%]
K-NN	130	10.3	7.7	2	16.3	***619***	846	232	5508	1	5	9
VAE	130	10.3	7.7	2	16.3	***561***	846	232	4871	1	5	1
RF	130	10.3	7.7	2	16.3	***478***	846	232	4898	1	5	16

Figure 16 – Demonstration of the imputation process of The Miles M-14 case

5. Conclusion

This research presents the validity of applying machine learning techniques to the aircraft conceptual design initial sizing problem. Conventional empirical methods of parameter estimation regarding the aircraft conceptual design have limitations in that only the insufficient size of the design space is explored. Also, the design result gets highly dependent on the experience and knowledge of a designer. This means that a lot of background knowledge of a user is required to get a satisfying design result. Furthermore, the calculation process is inefficient since each empirical formula requires different objectives. These limitations can be overcome by applying data-driven machine learning techniques.

Unlike using the empirical parameters or empirical relations, the data-driven approach enables the utilization of accumulated large datasets and enables the implementation of flexible input and output settings. However, the data-driven approach could not be used in aircraft design processes because existing aircraft data are mostly incomplete and heterogeneous. Additional advantages are retained by applying the data-driven machine learning models. Furthermore, the use of the data-driven machine learning models is advantageous since the models can read the hidden correlations among the parameters. The existing solvers can merely calculate the optimal values for each parameter, but deep learning models can emulate the decision-making process done by a human intelligence, and estimate the combinations of the design parameters that are not only optimal but also feasible for certain realistic reasons.

Three machine learning models – kNN, VAE, and RF – are utilized for the imputation of real-world incomplete aircraft data. kNN, VAE, and RF are capable of handling incomplete data that is randomly missing. To process heterogeneous data, both models have particular tactics. kNN separately processes regression and classification, while VAE and RF process the whole data simultaneously. kNN utilizes information from the nearest neighboring samples with an incomplete sample, and RF sets an appropriate criterion for each decision-making instant. VAE divides the distribution information of the missing and observed attributes and processes them separately.

VAE tends to be more accurate than kNN since it trains with the whole data simultaneously while kNN goes through the sequential process. RF has the opposite tendency regarding the regression and classification accuracies compared to kNN and VAE. Furthermore, from the user's perspective, VAE and RF are more convenient to use than kNN. Once the VAE and RF training results are saved, it is not necessary to carry the training data afterward. As the model or the data becomes complicated, VAE and RF implication would be much more convenient than kNN.

In conclusion, this study presented the validity of applying machine learning techniques to the initial sizing of the aircraft conceptual design problem as preliminary research. So far, this study suggested point estimations, but it will be able to suggest the feasible ranges of the design parameters and the uncertainties, making it more reasonable using the data-driven machine learning tactics.

6. Contact Author Email Address

Corresponding author: kjyee@snu.ac.kr

7. Copyright Statement

References

[1] Raymer D. *Aircraft design: a conceptual approach.* American Institute of Aeronautics and Astronautics, Inc., 2012.

[2] Moore M. ACSYNT *Aircraft Synthesis Program− User's Manual.* Systems Analysis Branch, NASA Ames Research Center, 1990.

[3] Ardema D. *Analytical fuselage and wing weight estimation of transport aircraft.* NASA Technical Memorandum, 1996.

[4] Liu M, Shi J, Cao K, Zhu J and Liu S. *Analyzing the training processes of deep generative models.* IEEE transactions on visualization and computer graphics, vol. 24, no. 1, pp. 77-87, 2017.

[5] Peyada N and Ghosh A. *Aircraft parameter estimation using neural network based algorithm.* AIAA atmospheric flight mechanics conference, pp. 5941, 2009.

[6] Janes Global UK Limited. *All The World's Aircraft: In Service Yearbook 20/21.* Janes Global UK Limited, 2020.

[7] Jauberthie C, Bournonville F, Coton P and Rendell F. *Optimal input design for aircraft parameter estimation.* Aerospace science and technology, vol. 10, no. 4, pp. 331-337, 2006.

[8] *Barnard J and Meng L. Applications of multiple imputation in medical studies: from AIDS to NHANES.* Statistical methods in medical research, vol. 8, no. 1, pp. 17-36, 1999.

[9] Schafer L and Graham W. *Missing data: our view of the state of the art.* Psychological methods, vol. 7, no. 2, p. 147, 2002.

[10] Fix E and Hodges L. *Discriminatory analysis. Nonparametric discrimination: Consistency properties.* International Statistical Review/Revue Internationale de Statistique, vol. 57, no. 3, pp. 238-247, 1989.

[11] Islam M, Iqbal H, Haque R and Hasan K. *Prediction of breast cancer using support vector machine and K-Nearest neighbors.* IEEE Region 10 Humanitarian Technology Conference (R10-HTC), pp. 226-229, 2017.

[12] Li M, Xu H, Liu X and Lu S. *Emotion recognition from multichannel EEG signals using K-nearest neighbor classification.* Technology and health care, vol. 26, no. S1, pp. 509-519, 2018.

[13] Triguero I, García-Gil D, Maillo J, Luengo J, García S. *Transforming big data into smart data: An insight on the use of the k-nearest neighbors algorithm to obtain quality data.* Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, vol. 9, no. 2, p. e1289, 2019.

[14] Kohli S, Godwin T and Urolagin S. *Sales prediction using linear and KNN regression.* Advances in machine learning and computational intelligence, pp. 321-329, 2021.

[15] Kara Z, Laksaci A, Rachdi M nd Vieu P. *Data-driven kNN estimation in nonparametric functional data analysis.* Journal of Multivariate Analysis, pp. 176-188, 2017.

[16] Ho T and Yu W. *Chiller system optimization using k nearest neighbour regression.* Journal of Cleaner Production, 303, 127050, 2021.

[17] Beretta L and Santaniello *A. Nearest neighbor imputation algorithms: a critical evaluation.* BMC medical informatics and decision making, vol. 16, no. 3, pp. 197-208, 2016.

[18] Falkowski J, Hudak T, Crookston L, Gessler E, Uebler H and Smith M. *Landscape-scale parameterization of a tree-level forest growth model: a k-nearest neighbor imputation approach incorporating LiDAR data.* Canadian Journal of Forest Research, vol. 40, no. 2, pp. 184-199, 2010.

[19] Chiu C, Selamat A and Krejcar O. *Infilling missing rainfall and runoff data for Sarawak, Malaysia using Gaussian mixture model based K-nearest neighbor imputation.* International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems, pp. 27-38, 2019.

[20] Goodfellow I, et al. *Generative adversarial nets*. Advances in neural information processing systems, vol. 27, 2014.

[21] Kingma P and Welling M. Auto-encoding variational bayes. arXiv preprint, arXiv:1312.6114, 2013.

[22] Yoon J, Jordon J and Schaar M. *Gain: Missing data imputation using generative adversarial nets*. International conference on machine learning, PMLR, pp. 5689-5698, 2018.

[23] Nazabal A, Olmos M, Ghahramani Z and Valera I. *Handling incomplete heterogeneous data using vaes*. Pattern Recognition, 107, 107501, 2020.

[24] Quinlan J. Induction of decision trees. *Machine learning,* vol. 1, no. 1, pp. 81-106, 1986.

[25] Salzberg, Programs for machine learning by j. ross quinlan, morgan kaufmann publishers, inc., 1993.

[26] Breiman L, Friedman JH, Olshen RA and Stone CJ. Classification and Regression Trees. *CRC Press*, 1984.

[27] Linero A. Bayesian regression trees for high-dimensional prediction and variable selection. *Journal of the American Statistical Association,* vol. 113, no. 522, pp. 626-636, 2018.

[28] Sanchez-Pinto N, Venable R, Fahrenbach J and Churpek M. Comparison of variable selection methods for clinical predictive modeling. *International journal of medical informatics,* vol. 116, pp. 10-17, 2018.

[29] Gregorutti B, Michel B and Saint-Pierre P. Correlation and variable importance in random forests. *Statistics and Computing*, vol. 27, no. 3, pp. 659-678, 2017.

[30] Chandrasekar P, Qian K, Shahriar H and Bhattacharya P. Improving the prediction accuracy of decision tree mining with data preprocessing. *2017 IEEE 41st Annual Computer Software and Applications Conference (COMPSAC), IEEE*, 2017.

[31] Liu C, Hu Z, Li Y and Liu S. Forecasting copper prices by decision tree learning. *Resources Policy,* vol. 52, pp. 427-434, 2017.

[32] Poulos J and Valle R. Missing data imputation for supervised learning. *Applied Artificial Intelligence,* vol. 32, no. 2, pp. 186-196, 2018.

[33] Breiman L. Random forests. *Machine learning,* vol. 45, no. 1, pp. 5-32, 2001.

[34] Stekhoven J and Bühlmann P. MissForest—non-parametric missing value imputation for mixed-type data. *Bioinformatics,* vol. 28, no. 1, pp. 112-118, 2012.