Available online at www.ijournalse.org

Emerging Science Journal

(ISSN: 2610-9182)

Vol. 8, No. 3, June, 2024

Product Design Cost Estimation for Make-to-Order Industry: A Machine Learning Approach

Anas Ma'ruf [1*], Raden A. C. Leuveano [2], Rizky Utama [1]

[1] *Faculty of Industrial Technology, Bandung Institute of Technology, Bandung, Indonesia.*

[2] *Faculty of Industrial Engineering, Universitas Pembangunan Nasional Veteran Yogyakarta, Yogyakarta, Indonesia.*

Abstract

This research addresses the need for accurate design cost estimation in the Make-To-Order (MTO) industry. The complexity of product customization is key to differentiation. While many studies focus on manufacturing cost estimation, few explore design cost estimation. To improve the accuracy of design cost estimation, this research proposes a new cost driver based on design features available in Computer-Aided Design (CAD) data. The design feature is analyzed to the actual industry cost using machine learning methods, including Artificial Neural Networks (ANNs) and Support Vector Regression (SVR). The cost drivers identified as significant consisted of twenty-six 3D CAD features and four 2D CAD features. The results showed that the ANN models outperformed the SVR models in correctly estimating product design costs, as evidenced by the high R^2 values in the training and testing phases. The proposed method allows early identification of cost drivers, a significant advantage at the order initiation stage when detailed design features are often ambiguous. The novelty of this research is the use of 3D CAD technology for cost estimation, which quantifies costs based on product design complexity, providing valuable insights into the impact of design adjustments on costs early in the design process.

Keywords:

Cost Estimation;

3D-CAD; Machine Learning;

Artificial Neural Network;

Support Vector Regression.

Article History:

Received:	03	November	2023
Revised:	28	April	2024
Accepted:	04	May	2024
Published:	01	June	2024

1- Introduction

The increasing demand for product customization underscores businesses' need to promptly and accurately respond to consumer orders, directly impacting the costs disclosed to the customer. Estimating these costs hinges on appropriate product design and customer approval, making cost prediction a fundamental criterion in evaluating engineering design costs [1, 2]. Accurate cost prediction gives designers a comprehensive understanding of manufacturing costs throughout the design process and facilitates early modifications to minimize these costs [3]. Estimation plays a crucial role in designing product costs, as it directly impacts the competitiveness of a business [4, 5]. If costs are estimated to be too high, this may lead to uncompetitive pricing and a loss of market share. On the other hand, underestimating costs can result in reduced profit margins or even losses on individual orders. Therefore, accurate estimation is vital for identifying the optimal balance between meeting customer customization demands and managing design costs effectively.

The Make-To-Order (MTO) industry presents unique product design and cost estimation challenges. The connection between product design and cost estimation is particularly crucial in this industry due to the need for rapid and accurate cost prediction to efficiently fulfill bespoke requirements. However, the design costs in the MTO industry represent a significant and intricate aspect of production expenses. The design is critical given the bespoke nature of the MTO industry, where each product is uniquely tailored to meet individual customer requirements. Design teams invest time

[*] **CONTACT**: maruf@itb.ac.id

DOI: http://dx.doi.org/10.28991/ESJ-2024-08-03-022

and expertise in understanding customer specifications, creating prototypes, and utilizing sophisticated computer-aided design tools. These expenses encompass not only the conceptualization of the product but also considerations for production feasibility, material selection, and adherence to quality standards. Managing design costs in the MTO industry requires a delicate balance between delivering tailored solutions and optimizing the efficiency of the design process to ensure market competitiveness. Due to the complexity of the design process, several surveys of the MTO industry apply design estimation, which can be classified into three approaches: a) personal judgment [6], b) design time/effort [7], or c) a percentage value of total product cost [8, 9].

One approach to design cost estimation in the MTO industry involves relying on the personal judgment of experienced design engineers. This method leverages the expertise and experience of individuals involved in similar projects to make informed estimates of design costs. While this approach can be helpful, it may also introduce variability based on individual perspectives and experiences, making it necessary to calibrate and validate the estimates against historical data and actual costs. Another approach entails estimating design costs based on the time or effort invested by the design team in developing a customized product. By quantifying the hours spent on activities such as understanding customer specifications, creating prototypes, and utilizing Computer-Aided Design (CAD) tools, organizations can derive a cost estimate based on hourly rates and the expertise of the personnel involved. However, this approach must also account for the potential variation in the efficiency and skill levels of the design team members, which can impact the accuracy of the estimates. The third approach involves deriving design costs as a percentage of the total product cost. This method establishes a predetermined percentage of the overall production expenses encompassing design-related activities. While this approach provides a simplified and standardized method for estimating design costs, it is essential to periodically reassess the validity of the predetermined percentage to account for changes in technology, materials, and design complexity. In conclusion, selecting an appropriate approach to design cost estimation in the MTO industry requires a thorough understanding of the organization's capabilities, historical data analysis, and continuous validation against actual costs.

In this context, accurate estimation of design costs becomes even more crucial due to the unique nature of each order in the MTO industry. Striking a balance between meeting customer customization demands and efficiently managing design costs is a significant challenge for businesses in this industry. The most critical costs to address early on are product design costs because they are usually determined based on the complexity of the design and machining planning procedures [10]. Moreover, the product design phase must be completed first due to its significant influence on the materials and machine planning process [11]. This means that there is a need for a detailed analysis of the factors affecting the selection of a material/process combination compared to the others, as well as the overall production cost. The critical importance of defining the early design phase leads this study to attempt to predict design costs using design features based on a CAD model of products.

The research problems studied were identified to predict product design costs, as indicated in Figure 1. It was observed from an investigation that the cost determination method applied by the MTO industry has the potential to produce values that can either be higher or lower than the correct price. This can lead to a decline in customer trust when the estimate is too high and the company's gross profit is too low. This is associated with the nonconsideration of all the design features in the CAD model. Therefore, the main focus of this study is to incorporate design features when estimating product design costs.

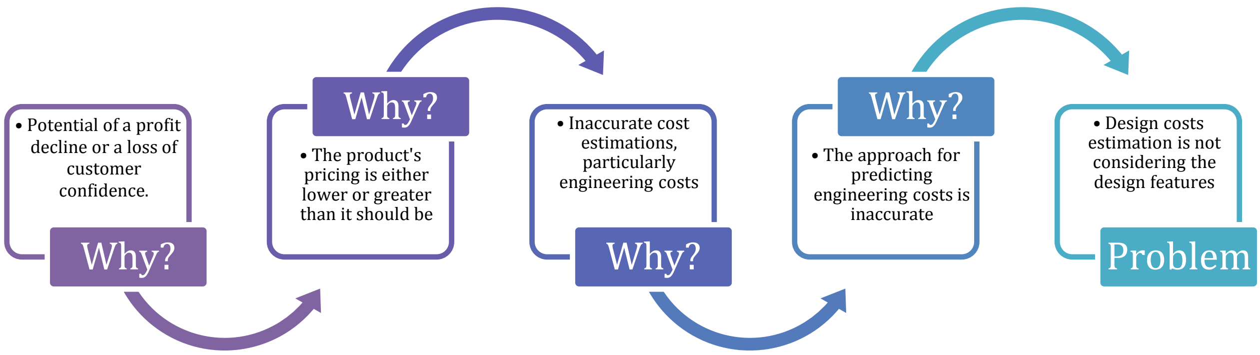


Figure 1. Fundamental issues related to the product design cost prediction in the MTO industry were investigated

In line with this focus, incorporating design features based on the CAD model of products can significantly enhance cost estimation accuracy in the MTO industry. By leveraging the detailed design features within the CAD model, including material selection, prototype creation, production feasibility, and adherence to quality standards, the predictive model can account for the nuanced complexities of customized product designs [9, 12]. This method addresses the shortcomings found in current design cost estimation methods, specifically in understanding the complexities of the customized needs of the MTO industry.

To address this, businesses must leverage advanced methods such as Machine Learning (ML) to improve cost estimation models. By integrating these into the design process, companies can gain valuable insights to optimize product designs, streamline manufacturing processes, and ultimately reduce manufacturing costs while improving customer satisfaction. ML algorithms can analyze and identify patterns within large datasets, allowing more precise cost estimations based on historical data and current design specifications [13].

ML approaches offer the advantage of capturing both linear and nonlinear data patterns, thereby enabling the generation of accurate estimation results [14, 15]. Various ML algorithms can be employed, such as neural networks, support vector machines, linear regression, decision trees, and random forests [16–18]. Each algorithm has its strengths and can be suitable for different aspects of design cost estimation. These algorithms can be used to analyze complex relationships between various design features and their respective costs effectively, ultimately improving the precision of cost estimations. By leveraging ML, businesses can develop models that consider the intricate interplay of multiple variables influencing design costs, leading to more reliable estimates.

This research proposes a systematic design cost estimation method for the early design stage of the MTO industry. Therefore, the objectives of this study are as follows:

- To analyze the effectiveness of adopting an ML approach, including an Artificial Neural Network (ANN) and Support Vector Regression (SVR), to estimate early product design costs using design features extracted from CAD models and actual company cost data.
- To develop an application to assist businesses in rapidly estimating the costs of designing products using the best prediction model.

Our research introduces an innovative approach to cost estimation, especially during the order acceptance phase of MTO operations, where product attribute scarcity typically occurs. Our study has two contributions. First, this study rigorously tests the efficacy of incorporating state-of-the-art ML techniques, specifically ANN and SVR, into early design cost prediction. This study relies on extracting key design features from CAD models, supplemented with actual cost data from the industry. By pitting ANN against SVR, our research uncovers intrinsic design-cost attributes. Second, we translated our research results into practical use by developing an application designed to empower the MTO industry with the ability to quickly and accurately estimate costs associated with product design. This application leverages the ability of the best-performing prediction models to optimize MTO operations.

The remainder of this paper includes Section 2, which focuses on reviewing the literature related to the applicability of the ML technique to early design cost estimation; Section 3 introduces the method applied to model ML estimations; Section 4 examines the outcomes of case studies involving the analysis of product design costs and analyses the performance of ML approaches; and Section 5 summarizes and discusses future works.

2- Related Works

Multiple approaches have been developed to estimate early design costs. In a study by Molcho et al. [19], these approaches were categorized into four main methods: expert opinion, parametric methods, analogical methods, and expert systems. The selection of the most suitable approach depends on the cost engineer's knowledge and the quantity of known product attributes/features. However, predicting costs during the early stages of product development remains challenging due to the limited information on conceptual product attributes [20]. Consequently, parametric methods are commonly favored; these methods involve creating simple cost functions based on the relationship between attribute values (e.g., production volume, product weight, machine size, cycle time) and costs [21].

The design process usually makes more information accessible, subsequently allowing the development of cost estimates for design features, manufacturing characteristics, and even a process plan. In this case, the fact that a sufficiently accurate cost function needs to be known by an engineer a priori is a significant difficulty of parametric methods [14]. Moreover, an unclear functional relationship between design features and costs frequently limits the ability of a defined function to adjust the parameters and fit the case data [20]. This means that a new cost-predicting relationship must consider additional design features instead of the previous parametric method.

Using ML algorithms in product design cost estimation offers a notable enhancement over traditional parametric methods [22]. By leveraging historical data and current design specifications, ML approaches can capture complex and nonlinear data patterns, allowing more precise cost estimations [14, 15]. Unlike parametric methods, ML approaches can adapt to changes in product specifications in real time, providing more accurate and comprehensive cost estimates. Additionally, the automation of cost estimation through ML reduces the time and resources required for manual calculations, enabling faster responses to customer orders and design modifications. One significant advantage of using ML for early design cost estimation in customized product design is its ability to adapt to each product's unique characteristics and complexities. Traditional parametric methods often struggle to account for the intricate interplay of multiple design variables and their respective impacts on costs. Conversely, ML excels at capturing these complex relationships and patterns, providing more accurate and reliable cost estimates for customized products.

The ML approach is Artificial Intelligence (AI) that can be applied to explore correlations between parameters using data without needing a predefined function with free parameters [20, 23]. This means that the primary advantages of this method include its relative simplicity in constructing its structures and its ability to simulate complicated nonlinear interactions and behaviors [24]. The ML approach can identify functional correlations between product design features and costs that are unknown to the cost engineer. This proves that different ML methods, such as ANNs, Support Vector Machines (SVMs), SVRs, K-nearest neighbors, and Bayesian methods, can resolve manufacturing problems such as estimation, detection, and classification [15, 23]. It is important to note that several studies have applied ML methods specifically for early cost estimation, as presented in Table 1, due to the numerous associated benefits.

Table 1. Previous Studies on Cost Estimation Using ML Method.

Researcher	Products	Input design features	ML method	Validation (Performance Comparison)
Cavalieri et al. [25]	Automotive products	Raw disk weight, number and type of foundry cores, material type, geometric shape, and disk type.	ANN	NN vs. parametric method using MAPE
Yeh & Deng [26]	Bending process and material of carbon steel pipe	Pipe diameter, pipe flange rate, no. of axes, the distance between the lending point and the end of the pipe, and no. of bends.	BPN and LS-SVM	NN vs. LS-SVM using MSE, MAPE, R^2
Loyer et al. [27]	Jet engine components	Span, mid-chord, number of blades, materials	GB, MLR, ANN, SVR, GAM	GB vs. MLR vs. ANN vs. SVR vs. GAM using MAPE and NRMSE
Leszczyński & Jasiński [28]	Product life cycle of induction motors	Rated power, rotational speed, rated current, no. of poles, shaft diameter, and net weight.	ANN	NN vs. parametric method using PE
Ning et al. [29]	Guide shaft, positioning guide shaft, guide-shaft bearing, roller, fixing ring, insertion pin, and metal gasket	The image files of 2D and voxel data of 3D parts.	CNN	2D CNN vs. 3D CNN using MAPE
Ning et al. [30]	Guide shaft, metal gasket, guide-shaft bearing, roller, fixing ring, and cantilever pin	Voxel of the 3D part-machining processing features, such as through-hole, diameter, depth, and cylinder.	CNN, SVM, and BNN	CNN vs. SVM vs. BNN using MAPE
Bodendorf & Franke [14]	Wheel products	Product lifetime, location of production, geometry, etc.), material (e.g., quantity/weight and price), process parameters (e.g. cycle time, melt weight), assembly steps, the process step (e.g. casting), surcharge rates (e.g. process production step costs), work activities, tooling invests, and time.	KNN, LR, SVR, NN, DT, AdaBoost Ensemble	KNN vs. LR. Vs. SVR vs. NN vs. DT vs. AdaBoost Ensemble using MSE, RMSE, and R^2
Kurasova et al. [13]	Furniture products	Item measurement, material data, operational data, labor data, batch size, and manufacturing complexity.	LR, DT, KNN, NN	LR vs. DT vs. KNN vs. NN using RMSE and R^2
Yoo & Kang [31]	CNC machined parts (e.g., pockets, slots, and holes)	Voxelized 3D CAD model (pockets, slots, and holes), costs, materials, and volume data.	CNN	RMSE and MAPE
Bodendorf et al. [32]	Circuit boards	Input image (Width, length, color)	CNN	AP and MAPE
Zhang et al. [33]	CNC machined rotary parts (e.g., turbine engine's case, shaft, and disc)	Geometrical, machining feature type, precision information	ConvGNN	MSE, MAE RMSLE
Klocker et al. [34]	Plastic injection molding parts	Weight, material, cavity, cycle time, labor time, tool costs, and lot size	DT, RF, GB, ANN	DT vs. RF vs. GB vs. ANN using MAE, MSE, RMSE, and MAPE
Hammann [35]	Automotive products	Total of 461 design features and the corresponding material costs	ANN, DT, GB, CBR, SVR, ELR, LAR	MAPE, NRMSE, EVS
Rapaccini et al. [18]	Components of Engineer-to-Order Products (ETOPs)	Qualitative features (material, supplier, etc.) and quantitative features (weight, length, number of items, etc.)	RF	MAPE
This study	MTO or Customized products	3D CAD features (point, line, arc, etc.)	ANN, SVR.	R^2
Note on machine learning approach	DT: Decision Tree, KNN: K-Nearest-Neighbors, LR: Linear Regression, ANN: Artificial Neural Network, SVM: Support Vector Machine, SVR: Support Vector Regression, GAM: Generalized Additive Models, LS-SVM: Least Squares Support Vector Machines; BNN: Backpropagation Neural Network, CNN: Convolutional Neural Network, RF: Random Forest, GB: Gradient Boosting, CBR: Case-based Reasoning; ELR: Elastic net Regression; LAR: Lasso Regression			
Note on performance measure	MAPE: Mean Absolute Percentage Error, MSE: Mean Square Error, MAE: Mean Absolute Error, RMSE: Root Mean Squared Error, PE: Percentage Error, R^2: The determinant coefficient, RMSLE: Root Mean Squared Log Error; NRMSE: Normalized Root Mean Square Error; EVS: Explained Variance Score; AP: Average Precision			

Based on Table 1, Cavalieri et al. [25] analyzed parametric and ANN models for estimating production costs in the automotive industry, underlining the significance of early cost management in product development. Although neural networks offer a better balance between accuracy and development cost, parametric models provide more precise data interpretation, which is crucial for design optimization during the early stages. Yeh & Deng [26] introduced a product cost estimation model for the entire product life cycle utilizing ML approaches such as Back-Propagation Neural networks (BPNs) and Least Squares Support Vector Machines (LS-SVMs), which outperform traditional statistical models in terms of accuracy. The enhancement of LS-SVMs through data transformation techniques is explored to mitigate outlier issues in the cost database, using airframe structure manufacturing as a case study for validation. The goal is to provide a more precise and universally applicable cost estimation model that is beneficial for cost planning and control across various industries. Loyer et al. [27] evaluated the effectiveness of five statistical models, namely, gradient boosted trees and support vector regression, for estimating the manufacturing cost of jet engine components using actual industrial data and found that these recent techniques significantly outperform traditional methods such as multiple linear regression and ANN. Considering various factors, such as computational cost and interpretability, in model selection is suggested. This highlights the importance of using multiple models for comprehensive analysis and demonstrates machine learning's potential for providing accurate, scalable cost estimates during the early design phase.

Leszczyński & Jasiński [28] compared ANNs with parametric models for estimating product life cycle costs and found that ANNs could significantly reduce estimation errors for complex products. It also introduces a method using customer technical specifications for automatic cost estimation, enhancing efficiency and reducing engineer workloads, with empirical support from a Polish induction motor company's data. Ning et al. [29] explored the efficacy of using 2D and 3D Convolutional Neural Network (CNN) models for manufacturing cost estimation and reported that 3D CNNs outperform 2D CNNs in regression-based tasks due to their ability to handle voxel data at various resolutions and data volumes. This advancement in deep learning methods, mainly through 3D CNNs, offers significant potential for improving cost estimation practices within the manufacturing industry. Ning et al. [30] introduced a method using 3D CNNs for precise feature recognition in part cost estimation enhanced by SVM and back propagation neural network techniques for establishing accurate cost-feature relationships, with the latter providing superior estimations. This approach showcases high accuracy and speed in identifying part features and demonstrates the method's adaptability and the significant potential of back propagation neural networks in cost estimation applications.

Bodendorf & Franke [14] evaluated six ML algorithms for predicting automotive wheel costs in early design phases, finding that all the models demonstrated high precision and accuracy but tended to undervalue total costs. Despite this, the efficiency and value of ML in enhancing cost engineers' decision-making and outperforming traditional spreadsheet calculations are emphasized, highlighting the importance of quality and quantity in training data. Kurasova et al. [13] introduced an ML-based approach for early cost estimation in customized furniture manufacturing, leveraging historical data to streamline and expedite the estimation process, albeit necessitating substantial historical data for effective training. This method, in contrast to traditional parametric and regression analyses, not only promises more accurate early-stage cost predictions with less human intervention but also aims to speed up product market entry by efficiently handling the complexities of estimating costs for custom designs with limited initial information.

Yoo & Kang [31] developed a deep learning-based method for estimating manufacturing costs and visualizing machining features in 3D CAD models utilizing a 3D CNN model and 3D Grad-CAM. The method effectively identifies and differentiates CNC machining features and their complexities, offering designers actionable insights for cost reduction and thus enhancing the efficiency of redesign processes for those with limited knowledge of manufacturing costs. Bodendorf et al. [32] presented a novel approach for early cost estimation in manufacturing by applying deep learning techniques, specifically focusing on image recognition, regression, and autoencoding for circuit board cost analysis. The study demonstrated that deep learning models, validated with real-world data from an Original Equipment Manufacturer (OEM), enhance cost estimation accuracy, with object recognition-based methods outperforming autoencoding techniques, offering transferable insights for other cost estimation projects.

Zhang et al. [33] presented a novel approach for estimating the manufacturing costs of CNC machined parts at the design stage using a Convolutional Graph Neural Network (ConvGNN) that incorporates precision information. This method, which significantly enhances the practicality of cost estimations compared to traditional deep learning models, involves constructing an attribute graph for machining features, developing a specialized ConvGNN named the Cost Estimation Network (CEN), and employing a modified Grad-CAM process for transparent decision-making, thereby achieving superior results by factoring in precision details. Klocker et al. [34] explored the use of machine learning for improving the accuracy and efficiency of cost estimation during early product development stages through a case study on plastic molding part production at an industrial company. It was found that tree-based algorithms surpass neural networks in terms of accuracy, mainly when predicting manufacturing parameters for individual process steps, enhancing cost estimation precision, and contributing to faster product development cycles.

Hammann [35] investigated the effectiveness of machine learning and big data in product cost estimation for passenger cars, highlighting the superior predictive accuracy of ML algorithms over traditional methods for complex products with numerous parts and cost drivers. The study demonstrated that ML significantly enhances cost estimation precision—up to 3.5 times greater—with big data and identifies critical cost drivers across thousands of product configurations, offering valuable insights for cost management in early product development stages. Rapaccini et al. [18] developed and evaluated an ML-based early cost estimation framework for engineer-to-order products (ETOPs) in the oil and gas industry, addressing the gap in data-driven methodologies for cost estimation in industrial applications. Through action research with a large industrial company, this study demonstrates ML's ability to explore the relationships between early design choices and cost estimation, resulting in an effective cost estimation framework with iterative feature selection and guidelines for integrating ML into industrial contexts with limited ML knowledge.

Based on Table 1, our study marks a gap in the literature by concentrating on the unique challenges of cost estimation within an MTO company, contrasting with prior research that has focused primarily on mass production contexts. Unlike these earlier works that applied ANNs for cost estimation, our research not only adopts ML techniques for early-stage product design cost estimation in the realm of customization but also delves into a detailed analysis of 3D CAD-extracted features as the basis for our models. This study expands upon the work of previous studies by rigorously examining a broader range of input features from 3D CAD models, including both 3D and 2D geometric features. This nuanced approach allows us to comprehensively explore the impact of CAD-based design features on design costs. Furthermore, our comparative analysis of ML methods, with a specific emphasis on performance validation using R^2 values, provides new insights into selecting the most effective technique for cost estimation in an MTO setting, thereby offering a novel contribution to the field.

3- Model Development

This study aimed to assess the applicability of ML approaches for cost estimation during the early phases of product design in the MTO industry for customized products. The focus is to ensure the possibility of using CAD models for products to predict product design costs. The new contribution of the proposed method in developing an estimation model is elaborated upon in the four-phase methodology, as shown in Figure 2.

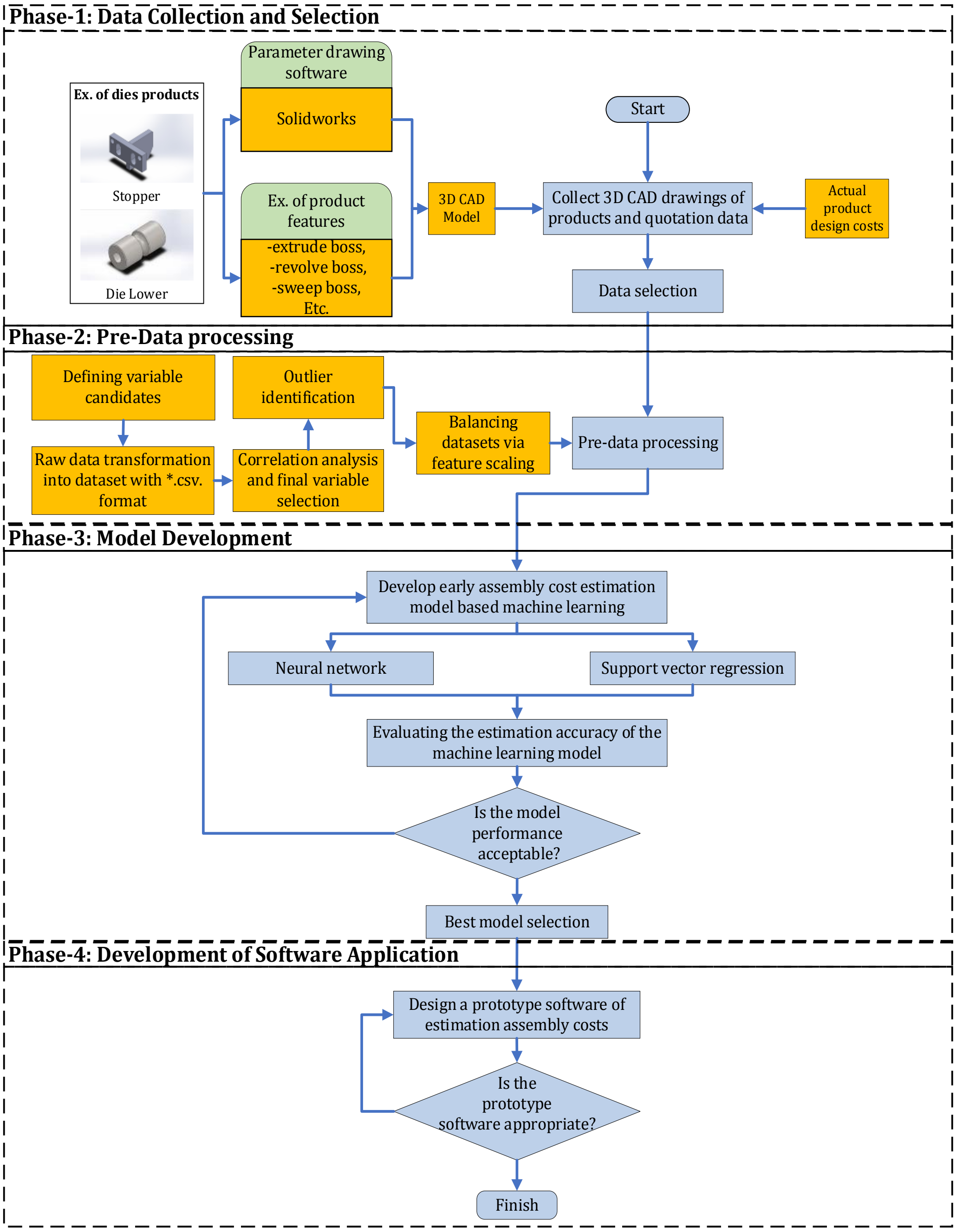


Figure 2. Methodology to predict the product design cost

3-1- Phase 1: Data Collection and Selection

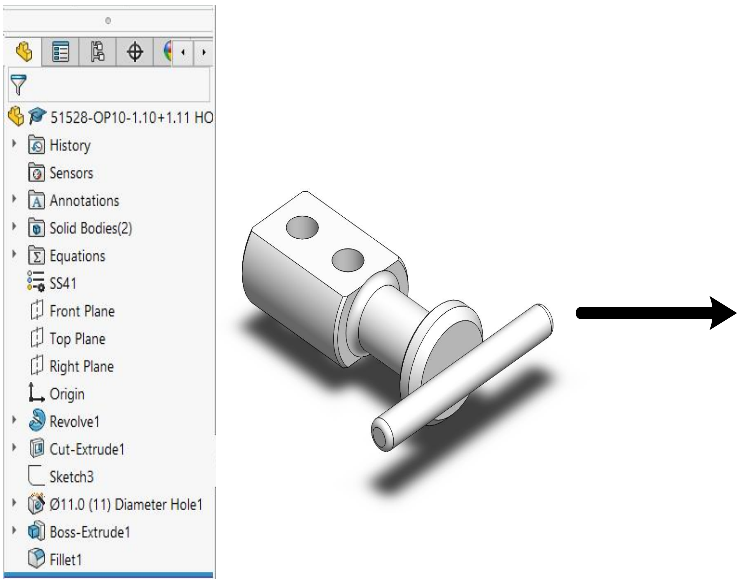
The first step was to collect historical data, which involved applying order processing for the MTO company, comprising three recent years of data as the sample for this study. The dataset consists of information on the CAD model, which contains design features and historical data on actual design costs stored by the company. It is important to note that the data for the CAD model's features was determined while the design costs were recorded in spreadsheet format.

The difference in these formats requires the application of a technique to merge the data into a single format to ensure a more straightforward automatic data transformation process. Therefore, the design cost data were incorporated into the CAD model, after which the irrelevant data were selected and eliminated. There were 486 different historical data records, with 36 entries containing irrelevant features removed. The design cost prediction model was developed using 450 raw order processing data records.

3-2- Phase 2: Pre-Data processing

The preprocessing phase is necessary since the data collected cannot be used directly to develop the model. The first step in pre-data processing is to identify candidate variables. The independent variables in the CAD model are features used in the 3D modeling process. It is important to note that 26 features are often used in creating CAD models, including four features for 2D, which are points, lines, arcs, and ellipses; twenty-two features for 3D, such as revolve boss, sweep boss, loft boss, extrude boss, boundary boss, revolve cut, sweep cut, loft cut, extrude cut, boundary cut, hole wizard, fillet, chamfer, linear pattern, circular pattern, mirror, rib, draft, shell, wrap, intersect, and dome. Moreover, the dependent variables are the actual costs associated with the product design.

The second step is to transform the raw data in the CAD model into a dataset in the form of a spreadsheet. The conversion process used Python in the CAD Application Programming Interface API. This procedure can decompose a CAD model into a number of features (independent variables), which are later stored in tabular format, as presented in Figure 3.



Feature	Number of Feature	Feature	Number of Feature
Point_Num	31	Boundary_Cut_Num	0
Line_Num	21	Fillet_Num	1
Arc_Num	3	Chamfer_Num	0
Ellips_Num	0	LPattern_Num	0
Boss_Extrude_Num	1	CirPattern_Num	0
Revolve_Num	1	Mirror_Num	0
Sweep_Num	0	Rib_Num	0
Loft_Num	0	Draft_Num	0
Boundary_Num	0	Shell_Num	0
Cut_Extrude_Num	1	Wrap_Num	0
Cut_Revolve_Num	0	Intersect_Num	0
Cut_Sweep_Num	0	Dome_Num	0
Cut_Loft_Num	0	Hole_Num	1

Cost : 6954

Figure 3. Raw data (CAD model) transformation into the dataset

Correlation analysis was subsequently conducted to determine the final selection of variables related to the number of features. This study applied the Spearman correlation coefficient because of its ability to analyze abnormally distributed datasets with linear or nonlinear correlations between variables, presented as a heatmap in Figure 4. Several variables had zero values, as indicated by the white line in the figure, because the features were used while developing the CAD models. These features were treated as independent variables because the models and software prototypes developed for design cost prediction in this study are intended for widespread usage by companies with MTO manufacturing systems. However, the Point_Num feature was excluded because it is strongly correlated with Line_Num and Arc_Num. This means that the model does not provide any additional information; rather, the model is more complex and biased. Finally, 25 features were used as independent variables to develop the model.

The next stage was used to identify the outlier data from the 450 raw order processing data records. The process was conducted using the Mahalanobis distance. These outliers were not eliminated because the company has processed the CAD model to enrich the variety of the models used as datasets.

The last stage in the data preprocessing process involves scaling the dataset features before they are input into the model. Various data scaling methods can be used, but Z-score standardization and min–max normalization can be performed. Model development after the data scaling process will be explained further in the next section.

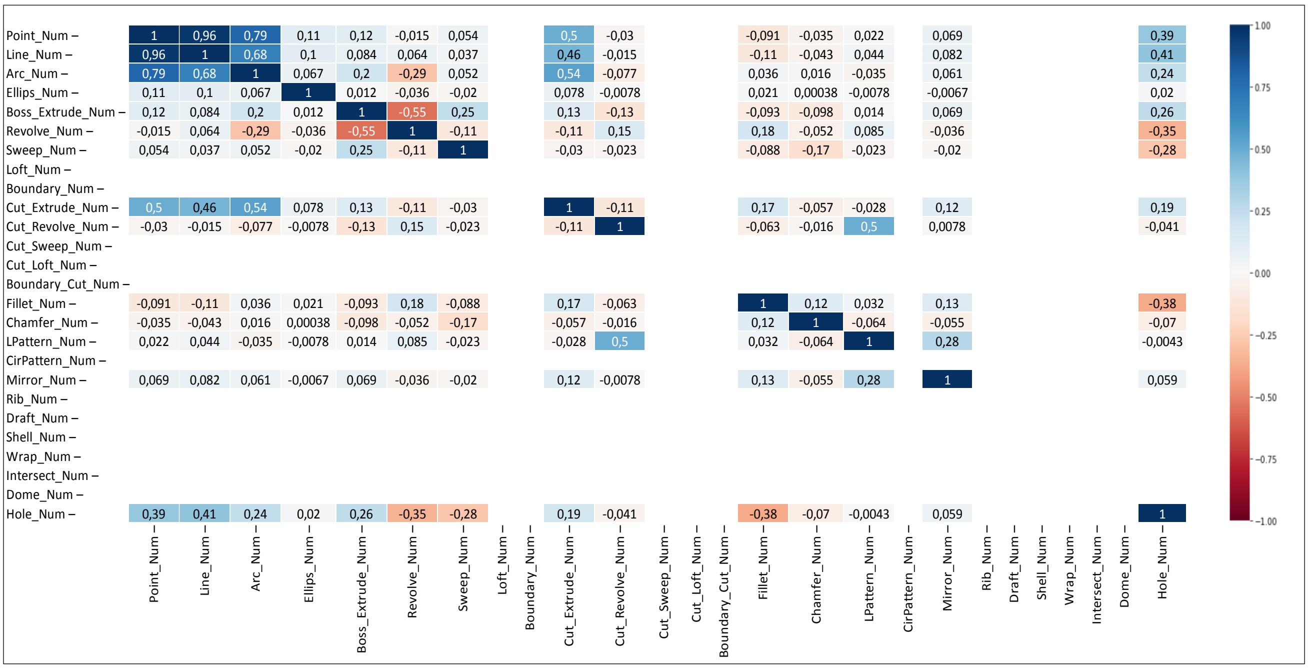


Figure 4. Correlation between features (Independent variables) using a heat map

3-3- Phase 3: Model Development

This subsection explains the development of two ML techniques, ANN and SVR models, for estimating the cost of a product's design.

ANN Model

An ANN is the mathematical representation of how neurons function in the brain [24]. A network is constructed of nodes (in ANNs, neurons) connected by layers and sets of layers to construct a network. An example of a multilayered ANN used in this study is indicated in the following Figure 5.

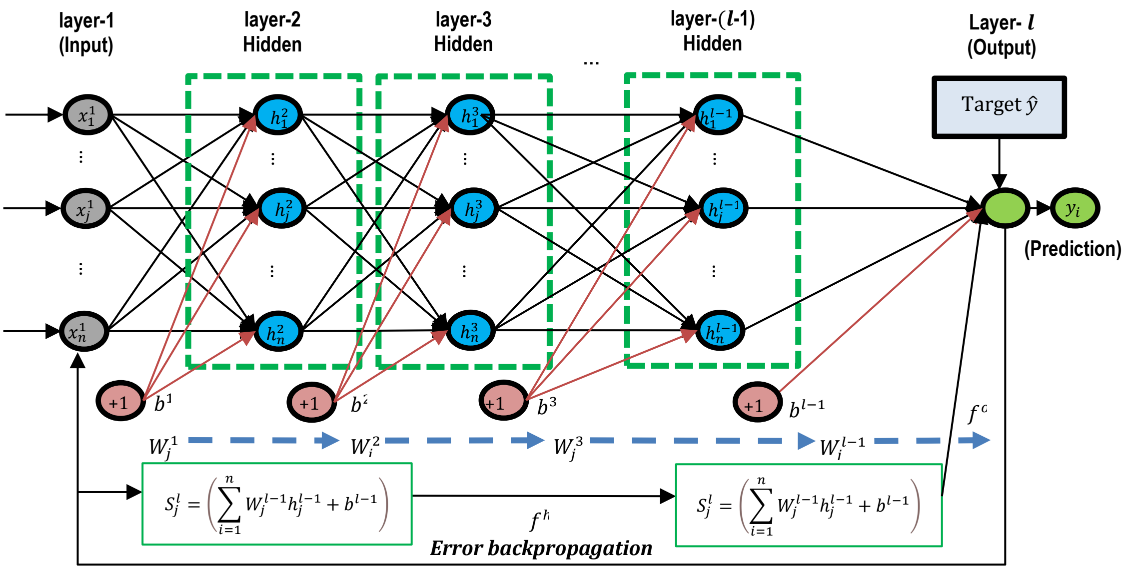


Figure 5. An example of ANN with a single input, multiple hidden layers, and a single output correlation between features (Independent variables) using a heat map

Figure 5 shows that the layers are classified into three types, namely, input, hidden, and output. Moreover, Backpropagation (BP) algorithms were employed in network configurations because feedforward (or activation propagation) and error BP are generally used to train networks [26].

During the feedforward stage, the input signal (preceding layer, l-1) to the hidden neuron nodes (the current layer, l) S_j^l was calculated as a function of the sum of the weighted inputs and bias, as expressed in Equation 1.

$$S_j^l = \left(\sum_{i=1}^{n} W_j^{l-1} h_j^{l-1} + b^{l-1}\right) \quad (1)$$

where $x_j^1 = h_j^{2-1}$ is the input vector defined for this study with 25 input variables x_1^1 to x_{25}^1 and W_j^{l-1} is the weight connecting each node j in the previous layer $(l-1)$ to node j in the current layer l. It is important to note that the bias in the prior layer is indicated by b^{l-1}. Subsequently, the output signal y_j^l for the neuron node j of the current hidden layer l and the output signal y_i for the output layer were obtained by passing the input signal S_j through the activation function employed in the calculation and propagation steps, as expressed in Equation 2 and Equation 3, respectively. The activation function used for each hidden layer is the rectified linear unit (ReLU) (f^h), while the output layer is linear (f^o).

$$y_j^l = f^h\left(S_j^l\right) = \max\left(0, S_j^l\right) \quad (2)$$

$$y_i = f^o\left(S_j^l\right) = S_j^l \quad (3)$$

The output signal y_j^l was also transmitted as an input signal to the neurons of the subsequent layer in multilayer ANNs. Therefore, the estimated output y_i was compared to the target output $\hat{y}_i$ at the end of the feedforward stage by calculating the loss function or error (E) via Equation 4.

$$E = \frac{1}{n}\sum_{l=i}^{n}(y_i - \hat{y}_i)^2 \quad (4)$$

The E value was backpropagated from the output layer to the input layer using a Bp training technique to minimize E by adjusting the link weights. The weight updating formula is provided in Equation 5.

$$W_j^{(l-1)^*} = W_j^{l-1} + \Delta W_j^{l-1} = W_j^{l-1} - \delta \frac{\partial E}{\partial W_j^{l-1}} \quad (5)$$

where $W_j^{(l-1)^*}$ is the new update weight connecting the input and hidden layers or the hidden layer and the output layer. In comparison, the current weight is W_j^{l-1}. Moreover, the error BP requires updating the gradient descent weights δ until a convergence criterion is met, indicating that the network output is nearly as desired.

In relation to E, this study also implemented the coefficient of determination (R^2) to calculate the accuracy of the ANN model based on the difference or error between the actual $\hat{y}_i$ and predicted y_i at its current position.

$$W_j^{(l-1)^*} = W_j^{l-1} + \Delta W_j^{l-1} = W_j^{l-1} - \delta \frac{\partial E}{\partial W_j^{l-1}} \quad (6)$$

The adoption of R^2 is due to its ability to capture the proportion of variation in actual values obtained by the ANN model. It provides a more accurate evaluation of the ANN model—the R^2 value ranges from 0 to 1, with 1 reflecting the closest possible relationship.

An experiment on the combination of hyperparameters is essential for determining the optimal designs of ANN models. This includes the number of neurons in every hidden layer, the number of hidden layers, the number of epochs, the batch size, and the learning rate [20]; Ning et al. [30] as an indication of the ANN architecture setup parameter presented in Table 2. The number of neuron and hidden layers is based on the rule of thumb defined by Heaton [36]. This research applies 2 to 5 hidden layers, 25 neurons in the input layer which corresponds to the number of design features and 1 neuron in the output layer. The number of epochs used is 1000 as a trial and error. The batch size is determined by the mini-batch gradient descent [37]. In most cases a good batch size is 32 or multiples [38]. The learning rate applies Adaptive Moment Estimation learning rate, set from 0.001 and gradually increases to 0.01 as proposed by Zulkifli [39].

Table 2. Hyperparameters used for ANN architectures.

Hyperparameters	**Range values**
Number of neurons in each hidden layer and number of hidden layers.	{(50, 50), (50, 25), (50, 50, 50), (50, 50, 25), (50, 50, 50, 50), (50, 50, 25, 13), (50, 50, 50, 50, 50), (50, 50, 25, 25, 25), (50, 50, 25, 25, 13), (50, 50, 25, 25, 13)}
Epoch	1000
Batch size	{32, 64, 128}
Learning rate	0.001 – 0.01

Note: (50, 50) represents two hidden layers (Layer 1: 50 neurons, Layer 2: 50 neurons)

SVR Model

SVR is a supervised ML technique usually applied to address estimation-based regression problems. It employs the fundamental concept of a SVR, a sparse kernel machine that performs regression using a few support vectors forming a hyperplane. However, the primary benefit of this approach is that it can handle linear and nonlinear regression estimations and curve fitting. To perform cost estimation in this study, we used the SVR model adopted from Zhang and O'Donnell [40]. Figure 6 shows a graphic representation of the SVR model under study.

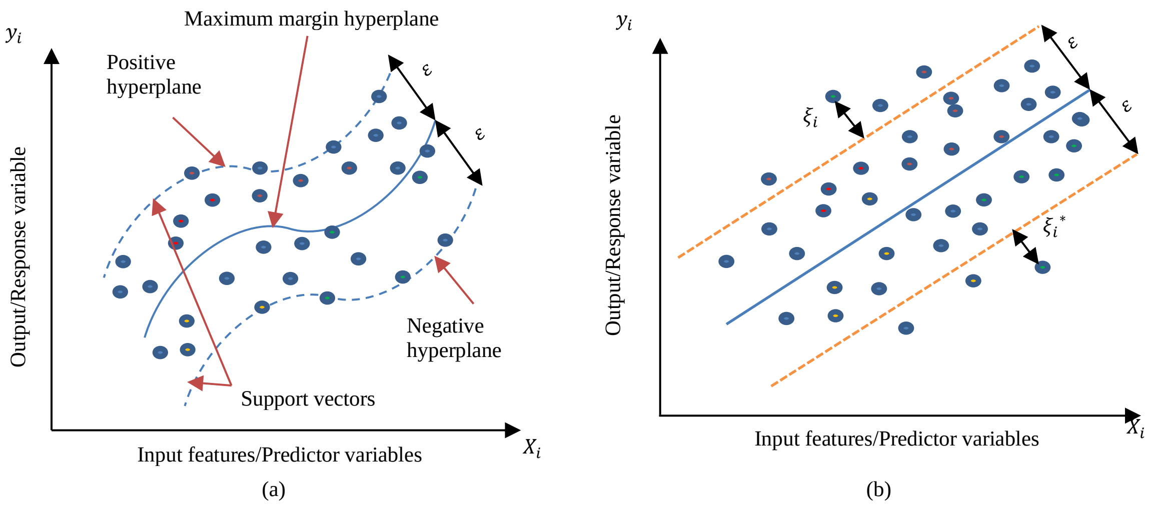


Figure 6. The representation of the nonlinear SVR model under investigation. a) A SVR maps all input data points using a mapping function on, ∅, where linear separation of the data is impossible to transform into a higher-dimensional kernel space, (b) a model that can be separated using a hyperplane (adapted from Zhang and O'Donnell [40]).

Figure 6 (a) shows that the SVR maps the input feature data into a higher-dimensional feature space using a nonlinear mapping function, $\emptyset$, to handle nonlinear data [41]. A function was later developed in this high-dimensional feature space under a linear function, as expressed in Equation 7.

$$y_i(X_i, w) = (w.\emptyset(X_i) + b), w \in \Re^d \tag{7}$$

Given the dataset $L = \{(X_i, \hat{y}_i)\}_{i=1}^{n} \in \Re^d \times \Re$,, i represents the number of training data points with input features X_i over space, $\Re^d, d$ represents the number of input features, w is a weight vector, b is a scalar bias, $\emptyset$ is the coefficient used for transforming the nonlinear problem into a linear problem, and y_i is associated with a single output.

Figure 6 (a) indicates the essence of SVR, which provides an ε-insensitive loss function to construct a hyperplane such that the predicted values (y_i) of the training samples deviate slightly from their actual (observed) values $(\hat{y}_i)$. Therefore, the slack variables $(\xi_i, \xi_i{}^*)$ and parameter C were introduced, as shown in Figure 6 (b), to minimize the ε-insensitive loss function (see Zhang and O'Donnell [40]). The problem of determining w and b in Equation 8 was reformulated as follows:

$$\text{Min } \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{N}(\xi_i + \xi_i{}^*) \tag{8}$$

Subject to $w.\emptyset(X_i) + b - y_i \leq \varepsilon + \xi_i{}^*, y_i - w.\emptyset(X_i) - b \leq \varepsilon + \xi_i, \xi, \xi^* \geq 0$

where C represents the positive constant or regularization parameter used to optimize the model errors and flatness. Figure 6 (b) indicates that the slack variables $(\xi_i, \xi_i{}^*)$ were introduced to prevent outliers or account for noisy data at the hyperplane boundary. Moreover, the optimization function of Equation 9 can be constructed in dual form by including the nonnegative Lagrange multiplier a_i, a_i^* for each observation X_i as follows:

$$\text{the Max L } (a_i, a_i^*) = \varepsilon\sum_{i=1}^{N}(a_i + a_i^*) + \sum_{i=1}^{N} y_i(a_i + a_i^*) - \frac{1}{2}\sum_{i=1}^{N}\sum_{i=1}^{N}(a_i - a_i^*) \times (a_j - a_j^*)K(X_i, X_i') \tag{9}$$

Subject to $\sum_{i=1}^{N}(a_i + a_i^*) = 0;\ a_i, a_i^* \in [0, C]$

The solution to the optimization problem 9 provides the unknown Lagrange multiplier, as expressed in the following Equation:

$$w = \sum_{k=1}^{n}(a_i - a_i^*)\ \emptyset(X_i) \tag{10}$$

Finally, the function y_i for predicting early product design costs was formulated as follows:

$$y_i(X_i) = \sum_{k=1}^{n}(a_i - a_i^*)\ K(X_i, X_i') + b \tag{11}$$

where $K(X_i, X_i') = \emptyset(X_i)\emptyset(X_i')$ and is known as the kernel function (KF). Moreover, the Karush–Kuhn–Tucker criteria can be used to generate the term b as follows:

$$b = y_i - (w.\emptyset(X_i)) - \varepsilon, a_i \in [0, C]$$
$$b = y_i - (w.\emptyset(X_i)) + \varepsilon, a_i^* \in [0, C] \quad (12)$$

The KF employed in Equation 12 for this study is the radial basis function (RBF), which is formulated as follows:

$$K(X_i, X_i') = Exp(-\gamma \|X_i - X_i'\|^2) \quad (13)$$

The bandwidth of the RBF was determined by gamma γ, while $\|X_i - X_i'\|$ is the Euclidean distance between X_i and X_i'. It is pertinent to state that there are two different kinds of γ: scale and auto. The value of γ for the scale type is $1/(d * \sigma(X_i))$, where d is the number of input features, $\sigma(X_i)$ is the variance of the input features, and $1/d$ is used as the γ value by the autotype.

The SVR model requires searching for an optimal setting of hyperparameter values (KF, ε, C, and γ), as presented in Table 3, to ensure accurate estimation performance. Moreover, the accuracy of the performance was assessed using the $the\ R^2$ value as expressed in Equation 6. It is also essential to state that the Python library scikit-SVM learn class was used to program the SVR model. This research applies the radial basis function (rbf) as the kernel function (KF). Radial kernels can be used to solve linear and non-linear problems and only require one parameter to be adjusted [42]. Two types of γ are applied: auto and scale. The hyperparameter of C and ε is set in an increasing value as presented in Table 3.

Table 3. Hyperparameters used for the SVR model

Hyperparameter	Range value
KF	{'rbf'}
γ	{'auto', 'scale'}
C	{0,1; 1; 10; 100; 1000; 10000}
ε	{0,1; 0,3; 0,5; 0,7; 0,9}

Model Validation

Grid search k-fold cross-validation was used in this study to validate the performance of different hyperparameter combination-based models. The k-fold cross-validation procedure was applied with k = 10 to split the dataset, which included 450 records, into 80% training and 20% test sets. Moreover, the performances of both the ANN and SVR models must be evaluated as previously stated by applying the statistical analysis R^2 in Equation 6 to determine the difference between their actual and predicted values.

3-4- Phase 4: Development of the Application Program for Early Product Design Cost Estimation

The best model in the previous phase was used to develop an application program with a graphical user interface (GUI) to ensure user friendliness. The application program has two main functions: the train model function and the cost estimation function.

4- Results and Discussion

ANN and SVR models were tested to predict the early product design cost. Table 2 showed 30 ANN models were generated for each min–max normalization and Z score standardization method used in the feature scaling process. Moreover, the hyperparameters listed Table 3 were used to create 60 SVR models for each method in a similar manner. To ease the analysis process, this study only presents the results of the five best models for each ML approach under both feature scaling methods of the dataset, as indicated in Tables 4 and 5, respectively.

Table 4. Results of the R^2 coefficient for the ANN model

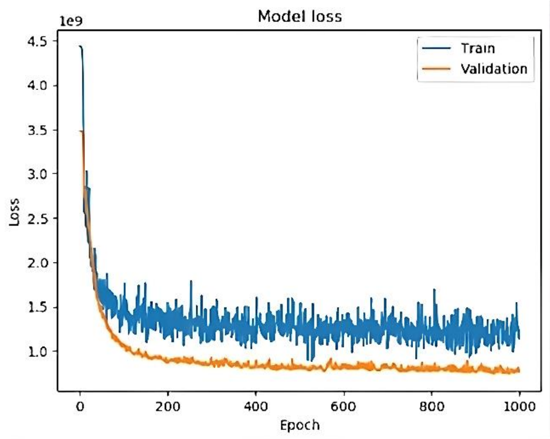
Model	Feature scaling dataset	Hidden layer structure	Batch size	Training score (R^2)	Test score (R^2)
ANN-1	Min–Max Normalization	(50, 50, 50, 50, 50)	128	0.7486	0.6542
ANN-2		(50, 50, 50, 50, 50)	64	0.7566	0.6401
ANN-3		(50, 50, 25, 25, 25)	32	0.7329	0.6374
ANN-4		(50, 50, 25, 25, 13)	32	0.6990	0.6344
ANN-5		(50, 50, 25, 25, 25)	64	0.7182	0.6312
ANN-31	Z score standardization	(50, 50, 25, 25, 25)	64	0.7341	0.5859
ANN-32		(50, 50, 25, 13)	128	0.6826	0.5809
ANN-33		(50, 50, 25, 25, 13)	32	0.7136	0.5804
ANN-34		(50, 50, 25, 13)	32	0.7046	0.5742
ANN-35		(50, 50, 25, 25, 13)	64	0.6850	0.5709

Table 5. Results of the R^2 coefficient for the SVR model

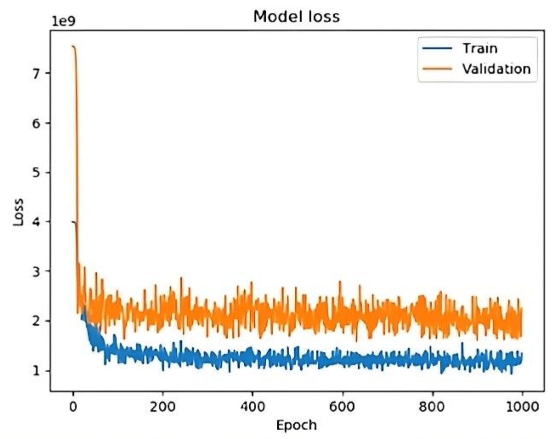
Model	Feature Scaling Dataset	*KF*	γ	C	ε	Training score (R^2)	Test score (R^2)
SVR-1	Min–Max Normalization	rbf	scale	10000	0.1	0.6521	0.6242
SVR-2		rbf	scale	10000	0.3	0.6521	0.6242
SVR-3		rbf	scale	10000	0.5	0.6521	0.6242
SVR-4		rbf	scale	10000	0.7	0.6521	0.6242
SVR-5		rbf	scale	10000	0.9	0.6521	0.6242
SVR-61	Z score standardization	rbf	auto	10000	0.1	0.6081	0.5938
SVR-62		rbf	auto	10000	0.3	0.6081	0.5938
SVR-63		rbf	auto	10000	0.5	0.6081	0.5938
SVR-64		rbf	auto	10000	0.7	0.6081	0.5938
SVR-65		rbf	auto	10000	0.9	0.6081	0.5938

Selecting the model with the highest R^2 in Table 4 and Table 5, it was found that ANN-1 and ANN-2 performed best for the test and training data, respectively. Moreover, the ANN-31 was used for Z score standardization. Further analysis revealed that ANN-1 outperformed both ANN-2 and ANN-31 in terms of the training data. It is essential to note that a good model should not exhibit a significant gap between test and training data. Thus, as shown in Table 4, ANN-1 was chosen as the best parameter setting for the ANN model. Moreover, SVR-1 to SVR-5 and SVR-61 to SVR-65 were observed to have the same accuracy despite the use of various parameters during the experiments since changes in the ε value do not affect the model's performance when C is constant. Moreover, the SVR-1 and SVR-61 models were selected as the best parameter settings under SVR even though all the models had the same R^2 value for each feature scaling method. It was also observed that ANN-1 has the highest R^2 value compared to SVR-1 and SVR-61, which led to its selection as the most accurate model for predicting product design cost.

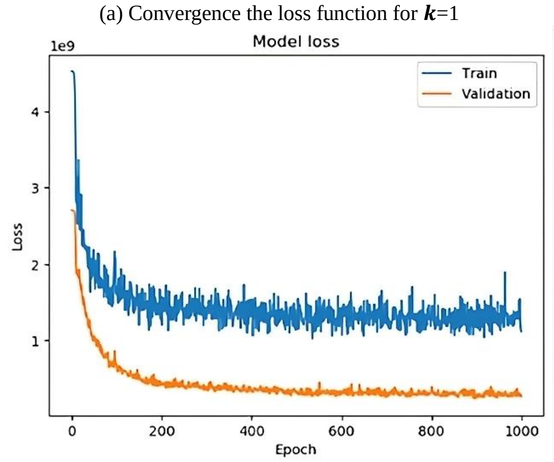
Interpretation of the learning curve is a technique for ensuring that the ANN-1 model is applied without overfitting or underfitting. Using the k-fold cross-validation approach with k=10, the model was developed, leading to the generation of 10 graphs with different folds. Figure 7 shows the estimation model's learning curve for ANN-1 based on the loss function (E). It is important to note that a lower loss function usually has better model performance. Figure 7 (a)-(j) shows that the training and test data graphs decrease over the first 100 epochs and remain constant for the next 1000 epochs. After which, 6 out of 10 iterations (k=10) produced a higher training graph than the test graph. This shows that the test set data are more easily predicted than the training set data. Moreover, Figure 7 (b) and (h) display a test graph above the training graph. This means that the data used for the training set are more accessible to predict than those used for the test set. These results showed that the model does not exhibit underfitting or overfitting.



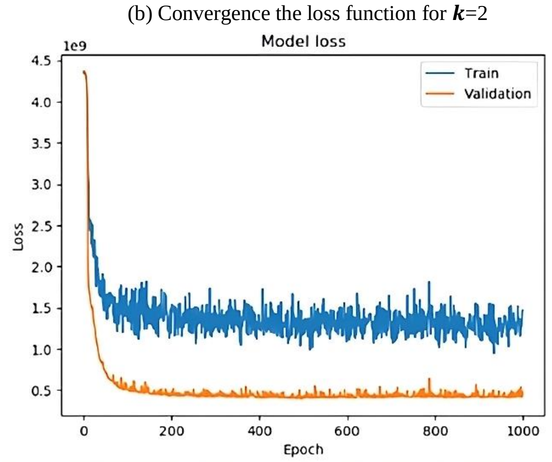
(a) Convergence the loss function for $\boldsymbol{k}$=1



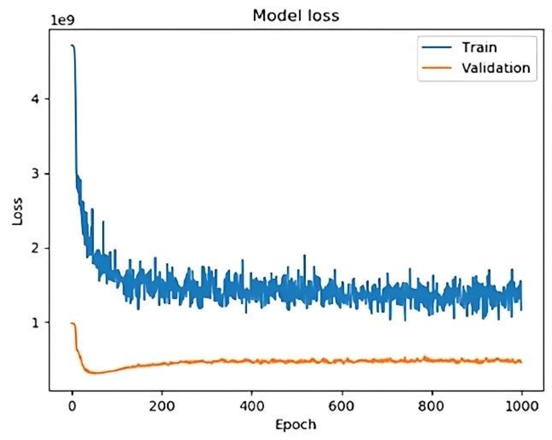
(b) Convergence the loss function for $\boldsymbol{k}$=2



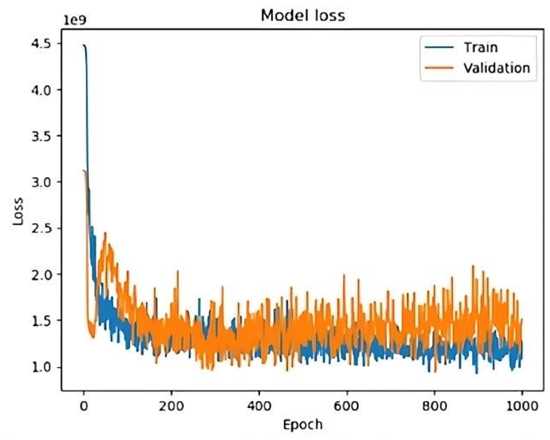
(c) Convergence the loss function for $\boldsymbol{k}$=3



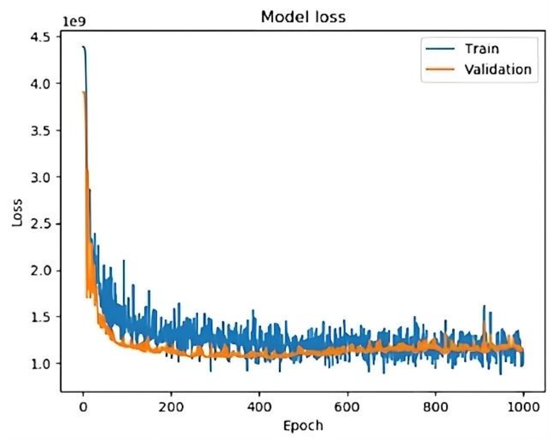
(d) Convergence the loss function for $\boldsymbol{k}$=4



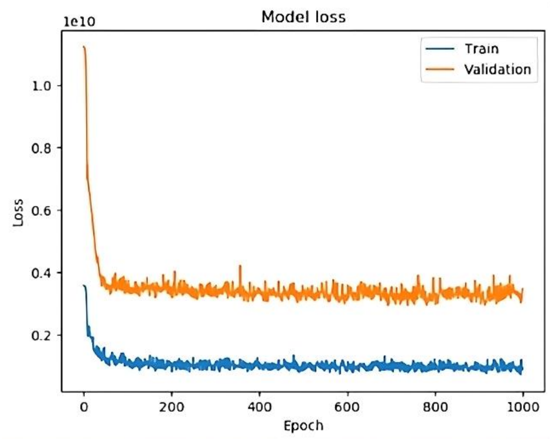
(e) Convergence the loss function for $\boldsymbol{k}$=5



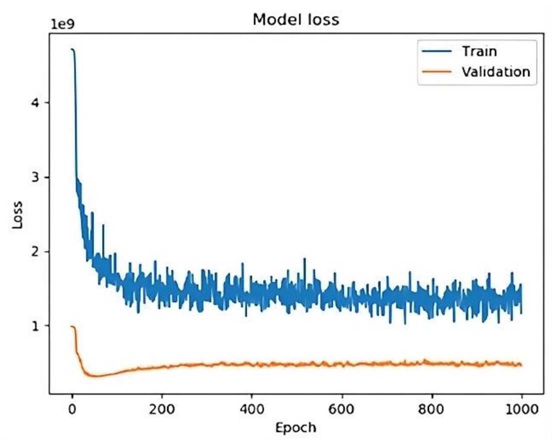
(f) Convergence the loss function for $\boldsymbol{k}$=6



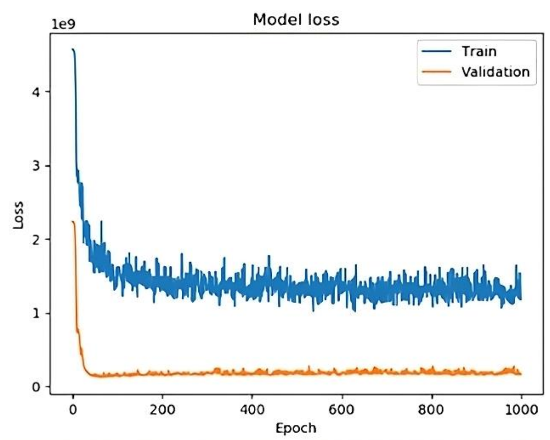
(g) Convergence the loss function for $\boldsymbol{k}$=7



(h) Convergence the loss function for $\boldsymbol{k}$=8



(i) Convergence the loss function for $\boldsymbol{k}$=9



(j) Convergence the loss function for $\boldsymbol{k}$=10

Figure 7. Learning Curve of ANN-1

The selected model also has a performance R^2 value of 0.65 for the test data. This indicates that the independent variables in the dataset can explain 65% of the variation in the dependent variable. This was followed by a validation procedure to determine the applicability of the designed model by the company in predicting the cost of the CAD model design for the subsequent order. The process involved comparing the actual and expected costs. The results obtained using the ANN-1 model are presented in Figure 8.

Figure 8 shows an example of ten CAD models tested, with errors ranging from 6% to 66% and an average error of 23%. Molcho et al. [19] proposed an early design cost estimation method and reported an average error of 35% in the actual cost. The cost drivers proposed by Molcho et al. [19] are similar to those of Kurasova et al. [13], which include part measurements, material type, machine type, batch size, setup complexity, required precisions, and design complexity. The Molcho et al. [19] and Kurasova et al. [13] models require additional information other than CAD. Also, a qualitative judgment is needed to evaluate design complexity, while this research utilizes the design features to consistently evaluate design complexity. It can be concluded that our proposed method for early cost estimation is concise due to direct reading of CAD data and without the requirement of qualitative judgment.

No. 3D Model	3D CAD Model's Name	Design Costs ($)	
		Actual	Predicted
1	Support Bracket	0.55	0.62
2	Stopper A	0.85	0.53
3	MISUMI-GP-MYP 32-140	1.41	1.10
4	Die Lower	1.27	0.44
5	Stopper B	0.81	0.61
6	Ejector	0.35	0.42
7	Rib	0.62	0.58
8	Guide Block	0.85	0.78
9	MISUMI-CP-AP10	0.50	0.61
10	Distance Block	0.57	0.52

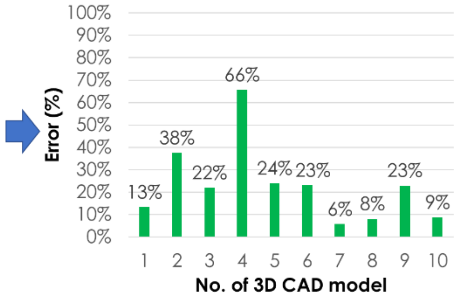


Figure 8. Comparison between actual and predicted costs based on ANN-1 model to validate new experimental data

Further analysis of variance revealed that the difference between the proposed model's actual and predicted design costs ranged from \$0.04 to \$0.84, and the difference for 8 out of the 10 examined CAD models was not statistically significant. This demonstrates that the estimation model accurately describes the design complexity of CAD models. The two CAD models with substantial variation were Stopper A and Die Lower, and their designs were not found to be excessively complicated. After checking with the company's engineers, it was discovered that they were both manufactured from more expensive materials. This means that the design costs were affected by the high cost of the materials. Moreover, the variation in the materials employed necessitates more significant material processing costs than other parts. By this finding, the proposed method can provide inherent information during product design.

It was concluded from this analysis that the design cost estimation model developed using corrected historical data has a reasonable degree of accuracy. The error rate recorded from the proposed method is acceptable for the MTO company. Hence, it is recommended that these materials be applied by the company. The application program developed in this study is presented in Figure 9. The user is presented with two main functions: the train model and estimated costs.

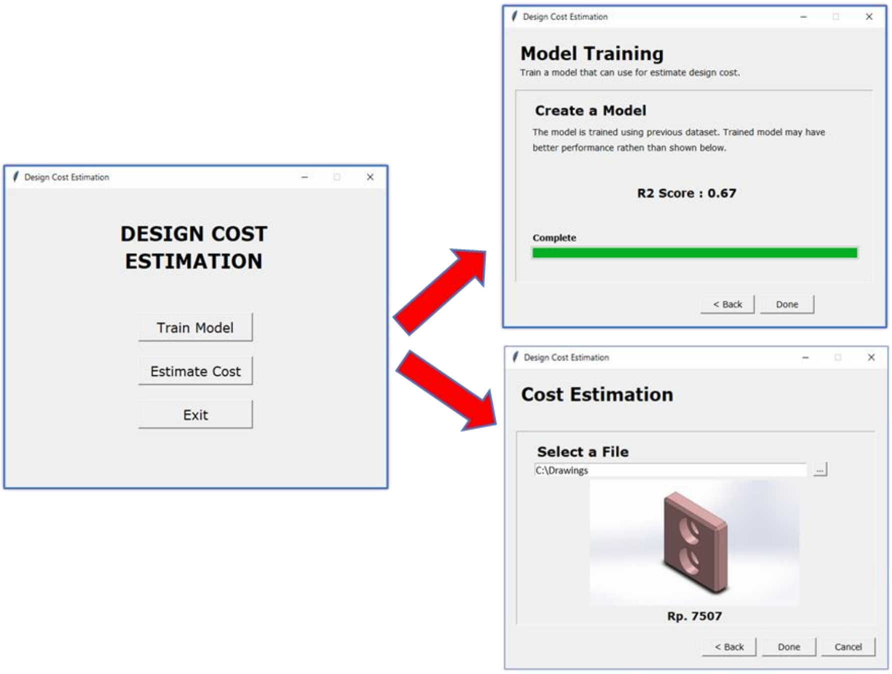


Figure 9. Prototype application to predict product design cost

The user is prompted to provide the CAD model as the training dataset in the training model function. This step is followed by executing the transformation data to convert the CAD model into a file containing the input features. After that, they are used as input in the ANN-1 model and trained using a dataset from the past. The R^2 of the training model is displayed before proceeding to the next phase, cost estimation. The cost estimation requires the user to select a CAD file, after which the predicted product design cost is calculated, as shown in Figure 9.

The change from the traditional design cost estimation method to the CAD model-based estimation using input features indicates improved product design cost estimation. The application program generally assists engineers in estimating product design costs. This approach provides a basis for order acceptance decisions during the initial business process of the MTO industry.

5- Conclusion

Industries operating in the MTO landscape are constantly challenged by the need to respond to customer orders quickly and accurately. This imperative demands the ability to predict product design costs precisely during the order acceptance phase, especially when manufacturing-related attributes are scarce. In response to this pressing need, our research has proposed a new method for product design cost estimation, an approach with practical implications for assisting MTO operations.

What distinguishes our approach is its unique ability to navigate complex design features and its close relationship with design costs, where comprehensive product attributes are severely lacking. This novel approach harnesses the power of advanced ML techniques, specifically ANN and SVR models, to perform the complex task of estimating product design costs with acceptable accuracy. Automatic design cost estimation is enabled by processing twenty-six 3D CAD features and four 2D CAD features as significant cost drivers.

Our comprehensive analysis revealed that ANN models consistently outperform SVR models in predicting product design costs. The superior performance of the ANN model, characterized by its accuracy and reliability, should be appreciated. These findings highlight the crucial advantages of our proposed methodology.

Furthermore, we developed a user-friendly application program as a core component of this research. This application has been field tested in an actual industrial environment, underscoring our proposed methodology's pragmatic feasibility and tangible effectiveness. To further improve the precision and applicability of our estimation model, future research studies should incorporate additional data relating to nongeometrical CAD data. Elements such as material, assembly process, and other nonstandard attributes indicate the potential to leverage accuracy in cost estimation.

6- Declarations

6-1- Author Contributions

Conceptualization, A.M., R.A.C.L., and R.U.; methodology, A.M., R.A.C.L., and R.U.; software, R.U.; validation, A.M. and R.U.; formal analysis, R.U.; investigation, A.M., R.A.C.L., and R.U.; resources, A.M.; data curation, R.U.; writing—original draft preparation, A.M. and R.A.C.L.; writing—review and editing, A.M. and R.A.C.L.; visualization, A.M. and R.A.C.L.; supervision, A.M.; project administration, A.M.; funding acquisition, A.M. All authors have read and agreed to the published version of the manuscript.

6-2- Data Availability Statement

The data presented in this study are available on request from the corresponding author.

6-3- Funding

This research was funded by the research, community services and innovation program of the Faculty of Industrial Technology, Bandung Institute of Technology.

6-4- Institutional Review Board Statement

Not applicable.

6-5- Informed Consent Statement

Not applicable.

6-6- Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this manuscript. In addition, the ethical issues, including plagiarism, informed consent, misconduct, data fabrication and/or falsification, double publication and/or submission, and redundancies have been completely observed by the authors.

7- References

[1] Quintana, G., & Ciurana, J. (2011). Cost estimation support tool for vertical high speed machines based on product characteristics and productivity requirements. International Journal of Production Economics, 134(1), 188–195. doi:10.1016/j.ijpe.2011.06.013.

[2] Niazi, A., Dai, J. S., Balabani, S., & Seneviratne, L. (2006). Product cost estimation: Technique classification and methodology review. Journal of Manufacturing Science and Engineering, 128(2), 563–575. doi:10.1115/1.2137750.

[3] Saravi, M., Newnes, L., Mileham, A. R., & Goh, Y. M. (2008). Estimating cost at the conceptual design stage to optimize design in terms of performance and cost. Collaborative Product and Service Life Cycle Management for a Sustainable World - Proceedings of the 15th ISPE International Conference on Concurrent Engineering (CE 2008), 123–130. doi:10.1007/978-1-84800-972-1_11.

[4] Bendul, J., & Apostu, V. (2017). An Accuracy Investigation of Product Cost Estimation in Automotive Die Manufacturing. International Journal of Business Administration, 8(7), 1-15. doi:10.5430/ijba.v8n7p1.

[5] Ievtushenko, O., & Hodge, G. L. (2012). Review of Cost Estimation Techniques and Their Strategic Importance in the New Product Development Process of Textile Products. Research Journal of Textile and Apparel, 16(1), 103–124. doi:10.1108/RJTA-16-01-2012-B012.

[6] Zhu, G. N., Hu, J., & Ren, H. (2020). A fuzzy rough number-based AHP-TOPSIS for design concept evaluation under uncertain environments. Applied Soft Computing Journal, 91, 106228. doi:10.1016/j.asoc.2020.106228.

[7] Holliman, A., Thomson, A., & Hird, A. (2021). Estimating design effort needs of product design projects using captured expert knowledge – A proposed method. Proceedings of the Design Society, 1, 1391–1400. doi:10.1017/pds.2021.139.

[8] Cai, X., & Tyagi, S. (2014). Development of a Product Life-Cycle Cost Estimation Model to Support Engineering Decision-Making in a Multi-Generational Product Development Environment. Journal of Cost Analysis and Parametrics, 7(3), 219–235. doi:10.1080/1941658x.2014.982403.

[9] Sajid, M., Wasim, A., Hussain, S., & Jahanzaib, M. (2018). Manufacturing feature-based cost estimation of cast parts. China Foundry, 15(6), 464–469. doi:10.1007/s41230-018-8084-4.

[10] Salmi, A., David, P., Blanco, E., & Summers, J. D. (2016). A review of cost estimation models for determining assembly automation level. Computers and Industrial Engineering, 98, 246–259. doi:10.1016/j.cie.2016.06.007.

[11] Dewhurst, P., & Boothroyd, G. (1988). Early cost estimating in product design. Journal of Manufacturing Systems, 7(3), 183–191. doi:10.1016/0278-6125(88)90003-9.

[12] Zhang, J., Chu, X., Simeone, A., & Gu, P. (2021). Machine learning-based design features decision support tool via customers purchasing data analysis. Concurrent Engineering Research and Applications, 29(2), 124–141. doi:10.1177/1063293X20963313.

[13] Kurasova, O., Marcinkevičius, V., Medvedev, V., & Mikulskienė, B. (2021). Early Cost Estimation in Customized Furniture Manufacturing Using Machine Learning. International Journal of Machine Learning and Computing, 11(1), 28–33. doi:10.18178/ijmlc.2021.11.1.1010.

[14] Bodendorf, F., & Franke, J. (2021). A machine learning approach to estimate product costs in the early product design phase: A use case from the automotive industry. Procedia CIRP, 100, 643–648. doi:10.1016/j.procir.2021.05.137.

[15] Dogan, A., & Birant, D. (2021). Machine learning and data mining in manufacturing. Expert Systems with Applications, 166(15), 114060. doi:10.1016/j.eswa.2020.114060.

[16] Balal, A., Jafarabadi, Y. P., Demir, A., Igene, M., Giesselmann, M., & Bayne, S. (2023). Forecasting Solar Power Generation Utilizing Machine Learning Models in Lubbock. Emerging Science Journal, 7(4), 1052–1062. doi:10.28991/ESJ-2023-07-04-02.

[17] Antonopoulou, H., Theodorakopoulos, L., Halkiopoulos, C., & Mamalougkou, V. (2023). Utilizing Machine Learning to Reassess the Predictability of Bank Stocks. Emerging Science Journal, 7(3), 724–732. doi:10.28991/ESJ-2023-07-03-04.

[18] Rapaccini, M., Cadonna, V. L., Leoni, L., & De Carlo, F. (2023). Application of machine learning techniques for cost estimation of engineer to order products. International Journal of Production Research, 61(20), 6978–7000. doi:10.1080/00207543.2022.2141907.

[19] Molcho, G., Cristal, A., & Shpitalni, M. (2014). Part cost estimation at early design phase. CIRP Annals - Manufacturing Technology, 63(1), 153–156. doi:10.1016/j.cirp.2014.03.107.

[20] Bode, J. (2000). Neural networks for cost estimation: Simulations and pilot application. International Journal of Production Research, 38(6), 1231–1254. doi:10.1080/002075400188825.

[21] Qian, L., & Ben-Arieh, D. (2008). Parametric cost estimation based on activity-based costing: A case study for design and development of rotational parts. International Journal of Production Economics, 113(2), 805–818. doi:10.1016/j.ijpe.2007.08.010.

[22] Manuguerra, L., Mandolini, M., Germani, M., & Sartini, M. (2023). Machine Learning for Parametric Cost Estimation of Axisymmetric Components. Proceedings of the Design Society, 3, 2485–2494. doi:10.1017/pds.2023.249.

[23] Bertolini, M., Mezzogori, D., Neroni, M., & Zammori, F. (2021). Machine Learning for industrial applications: A comprehensive literature review. Expert Systems with Applications, 175, 114820. doi:10.1016/j.eswa.2021.114820.

[24] Durodola, J. F. (2022). Machine learning for design, phase transformation and mechanical properties of alloys. Progress in Materials Science, 123(August 2020), 100797. doi:10.1016/j.pmatsci.2021.100797.

[25] Cavalieri, S., Maccarrone, P., & Pinto, R. (2004). Parametric vs. neural network models for the estimation of production costs: A case study in the automotive industry. International Journal of Production Economics, 91(2), 165–177. doi:10.1016/j.ijpe.2003.08.005.

[26] Yeh, T. H., & Deng, S. (2012). Application of machine learning methods to cost estimation of product life cycle. International Journal of Computer Integrated Manufacturing, 25(4–5), 340–352. doi:10.1080/0951192X.2011.645381.

[27] Loyer, J. L., Henriques, E., Fontul, M., & Wiseall, S. (2016). Comparison of Machine Learning methods applied to the estimation of manufacturing cost of jet engine components. International Journal of Production Economics, 178, 109–119. doi:10.1016/j.ijpe.2016.05.006.

[28] Leszczyński, Z., & Jasiński, T. (2020). Comparison of product life cycle cost estimating models based on neural networks and parametric techniques—a case study for induction motors. Sustainability, 12(20), 1–14. doi:10.3390/su12208353.

[29] Ning, F., Shi, Y., Cai, M., Xu, W., & Zhang, X. (2020). Manufacturing cost estimation based on a deep-learning method. Journal of Manufacturing Systems, 54, 186–195. doi:10.1016/j.jmsy.2019.12.005.

[30] Ning, F., Shi, Y., Cai, M., Xu, W., & Zhang, X. (2020). Manufacturing cost estimation based on the machining process and deep-learning method. Journal of Manufacturing Systems, 56, 11–22. doi:10.1016/j.jmsy.2020.04.011.

[31] Yoo, S., & Kang, N. (2021). Explainable artificial intelligence for manufacturing cost estimation and machining feature visualization. Expert Systems with Applications, 183, 115430. doi:10.1016/j.eswa.2021.115430.

[32] Bodendorf, F., Merbele, S., & Franke, J. (2022). Deep learning based cost estimation of circuit boards: a case study in the automotive industry. International Journal of Production Research, 60(23), 6945–6966. doi:10.1080/00207543.2021.1998698.

[33] Zhang, H., Wang, W., Zhang, S., Huang, B., Zhang, Y., Wang, M., Liang, J., & Wang, Z. (2022). A novel method based on a convolutional graph neural network for manufacturing cost estimation. Journal of Manufacturing Systems, 65, 837–852. doi:10.1016/j.jmsy.2022.10.007.

[34] Klocker, F., Bernsteiner, R., Ploder, C., & Nocker, M. (2023). A Machine Learning Approach for Automated Cost Estimation of Plastic Injection Molding Parts. Cloud Computing and Data Science, 4(2), 87–111. doi:10.37256/ccds.4220232277.

[35] Hammann, D. (2024). Big data and machine learning in cost estimation: An automotive case study. International Journal of Production Economics, 269, 109–137. doi:10.1016/j.ijpe.2023.109137.

[36] Heaton, J. (2015). Deep learning and neural networks. Machine Learning and Visual Perception. De Gruyter Brill, Berlin, Germany. doi:10.1515/9783110595567-010.

[37] Ruder, S. (2016). An Overview of Gradient Descent Optimization Algorithms. Ruder.io, Berlin, Germany. Available online: https://www.ruder.io/optimizing-gradient-descent/ (accessed on February 2024).

[38] Bengio, Y. (2012). Practical recommendations for gradient-based training of deep architectures. In Neural networks: Tricks of the trade: Second edition. Springer, Berlin, Germany.

[39] Zulkifli, H. (2018). Understanding Learning Rates and How It Improves Performance in Deep Learning. Medium, California, United States. Available online: https://towardsdatascience.com/understanding-learning-rates-and-how-it-improves-performance-in-deep-learning-d0d4059c1c10 (accessed on February 2024).

[40] Zhang, F., & O'Donnell, L. J. (2020). Support vector regression. Machine Learning, 123–140, Academic Press, Massachusetts, United States. doi:10.1016/b978-0-12-815739-8.00007-9.

[41] Tang, Z., Yin, H., Yang, C., Yu, J., & Guo, H. (2021). Predicting the electricity consumption of urban rail transit based on binary nonlinear fitting regression and support vector regression. Sustainable Cities and Society, 66(2), 1–14. doi:10.1016/j.scs.2020.102690.

[42] Laref, R., Losson, E., Sava, A., & Siadat, M. (2019). On the optimization of the support vector machine regression hyperparameters setting for gas sensors array applications. Chemometrics and Intelligent Laboratory Systems, 184, 22–27. doi:10.1016/j.chemolab.2018.11.011.