Optimization of Support Vector Regression Algorithm Using Simulated Annealing for Used Car Price Prediction

Dina Rizka Luviani*[1] , Rifqi Salman Hakim[2] , Wikan Haydarrahman[3]
[1,2]Institution/affiliation; addres, tel/fax of institution/affiliation
[3]Department of Computer Science, Semarang State University, Semarang
e-mail: ***[1]dinarizkalvn@students.unnes.ac.id,** [2]**rifqisalmanhakim28@unnes.ac.id,**
[3]**wikanhaydar23peb@students.unnes.ac.id,** [4]**floyuna@mail.unnes.ac.id**

Abstract

Used car price prediction requires an accurate method given the complexity of various price determining factors such as make, model, and vehicle condition. This research proposes the integration of Simulated Annealing optimization algorithm in the Support Vector Regression model to improve the accuracy of used car price prediction. The dataset used consists of 9 features, but in this study to predict the price of used cars, only 4 features are used that have a significant relationship with the prediction target (price). The evaluation results show that the Support Vector Regression (SVR) model with Simulated Annealing (SA) optimization has the best performance in predicting used car prices, with an MAE value of 786.10, MSE of 1,467,156.40, RMSE of 1211.26, and R² Score of 0.9655 compared to other models. The SVR model with SA provides higher prediction accuracy and is able to explain 96.55% of the data variability better.Abstract

***Keywords-Car**; Prediction; SVR; Regression; Simulated Annealing*

Abstract

Predicting used car prices requires accurate methods due to the complexity of various factors influencing the price, such as brand, model, and vehicle condition. This study proposes integrating the Simulated Annealing optimization algorithm with a Support Vector Regression (SVR) model to enhance the accuracy of used car price prediction. The dataset consists of 9 features, but only 4 features significantly related to the target prediction (price) are used in this study. Evaluation results show that the Support Vector Regression (SVR) model optimized with Simulated Annealing (SA) performs best in predicting used car prices, achieving a Mean Absolute Error (MAE) of 786.10, Mean Squared Error (MSE) of 1,467,156.40, Root Mean Squared Error (RMSE) of 1211.26, and an R² score of 0.9655 compared to other models. The SVR model with SA provides higher prediction accuracy and effectively explains 96.55% of the data variability.

***Keywords-Car**; Prediction; SVR; Regression; Simulated Annealing*

1. INTRODUCTION

D

ith the rapid development of the automotive industry, cars have become an important means of travel for many people[1][2]. Along with the growth of the market, used cars are also increasingly considered a common item and attract the attention of many people. The popularity of used cars is driven by more affordable prices compared to new cars[1],[3], this affordable price is what makes used cars nowadays in great demand, and also because they can be reached from various economic circles. Because there is a lot of interest and demand for buying and selling used cars today, many used car buying and selling transactions are carried out online[4]. With regard to that, accurate price prediction is one of the things that needs to be understood by buyers, and sellers[5],[6] with that understanding will also provide benefits for sellers[3]. The price of a used car is influenced by various factors such as brand, model, year of production, mileage, and physical condition[1],[7],[8] which makes it complex. This complexity makes the price determination process a challenge[2] and requires a more efficient method to accurately predict the price[9]. Used car pricing is often a complex process due to the various variables that affect the final value of the vehicle[7]. Data-driven prediction methods are a potential solution in determining used car prices more accurately. Machine learning models are one of the most widely used approaches, which can process various variables and patterns in data to produce more precise price estimates.

Some studies show that machine learning is able to provide more reliable predictions [10], [11], [12] than traditional methods, because these algorithms can learn from historical data and other complex factors [10]. Some studies that prove this include research conducted by [12] in this study using the random forest algorithm to predict the price of used cars, the results of the study state that by using random forest the accuracy value R^2 obtained is 77.2584%. In addition, research conducted by [13] discusses the same thing but in this study a comparison is made between 3 algorithms, namely the *Random Forest algorithm, Decission Tree,* and *Multiple Linear Regression*, from this study it is proven that of the 3 algorithms used *Random Forest* has the greatest accuracy of 94.10% followed by *Decission Tree* of 92.45% and the last 89.85% with *Multiple Linear Regression*.

Support Vector Regression (SVR) is used in this study because this algorithm has the advantage of handling complex data that is influenced by many factors, such as brand, fashion, year of production, and others. *Support Vector Regression* (SVR), one of the machine learning approaches developed by Vladimir N. Vapnik in 1995 [14]. SVR is able to cope with non-linear data [15], [16] SVR applies a kernel that allows modeling non-linear relationships in data [16], and has a good ability to model complex relationships between various variables, such as age, mileage, and used car prices. This is evidenced by research conducted by [17] In this study SVR was used to predict the price of palm oil, the results of this study prove that SVR is able to provide excellent prediction results of 98.83%, However, apart from that SVR requires optimization to find the best parameter values to prevent *overfitting* [18] because in this study the parameter values are still inputted manually to get the best results.

To improve the performance of used car price prediction models, an optimization approach to SVR parameters is required. *Simulated Annealing* is one of the optimization methods that can be applied. *Simulated Annealing* uses a metaheuristic approach inspired by the physical process of metal cooling to solve optimization problems [19], which works by exploring various solutions globally, and avoiding the trap of local solutions. The simulated annealing algorithm is very suitable to be used to find the best probability, this is proven by research conducted by [20], the results of this study obtained by applying the simulated annealing algorithm the accuracy obtained reached 98.66%. Therefore, in this study, a combination of the SVR algorithm as the main algorithm with the simulated annealing algorithm as an optimization algorithm will be carried out. By applying this algorithm to SVR, it is expected to find more optimal parameters, such as kernel and regulation value, so as to improve the accuracy of used car price prediction. This study aims to integrate *the Simulated Annealing* algorithm into the *Support Vector Regression* model to predict used car prices more accurately. With this optimization, it is expected that the model will be able to provide a more effective solution in determining the price of used cars.

2. RESEARCH METHODS

In this research, the *Support Vector Machine* algorithm approach is used which is optimized with the Simulated Annealing optimization algorithm. The flowchart below illustrates the systematic steps taken in the machine learning project to produce accurate and optimized predictive models.

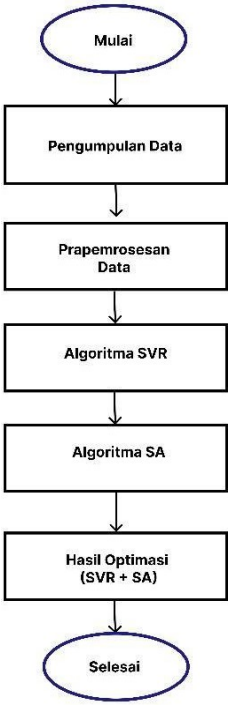


Figure 1 Research flowchart

The flowchart in **Figure 1** describes the stages of a modeling and optimization algorithm project. The process starts with *Data Collection*, which is the collection of relevant data, such as used car price datasets that contain specific information such as year, price, and other car characteristics. A dataset titled '100,000 UK Used Car Data set' was retrieved from kaggle for use in this research. In this dataset there are 13 csv files, one of which is "toyota.csv". This dataset contains information about used car prices, with several key attributes that affect the price of the car. This dataset has 9 attribute columns containing various features that affect car prices and 6,738 data entries. Once the data is obtained, *Data Preprocessing* is performed, which involves cleaning and transforming the data to make it ready for use by the model, including normalization, or variable encoding.

The next stage is *Modeling / Train Data with Algorithm*, where the initial model is trained using basic algorithms such as linear regression or SVR (Support Vector Regression). After the model is trained, the results are evaluated at the *Evaluation The First Model* stage using evaluation matrices such as *Mean Squared Error* (MSE), *Root Mean Squared Error* (RMSE), and R^2 *Score*. If the model has not provided satisfactory results, the process continues with *Optimization First Model Using Optimization Algorithm*, where the model is optimized using optimization algorithms such as *Simulated Annealing Optimization*. After optimization the model and final results are saved in *Save Model* and *Final Result*. The process is closed with the *End* step, which signifies that the model is ready to be used for further implementation.

2.1. Data Collection and Preparation

The dataset titled '100,000 UK Used Car Data set' was retrieved from kaggle for use in this research. This dataset contains 13 csv files, one of which is "toyota.csv". This dataset contains information about used car prices, with several key attributes that affect the price of the car. This dataset has 9 attribute columns containing various features that affect car prices and 6,738 data entries. https://www.kaggle.com/datasets/adityadesai13/used-car-dataset-ford-and-mercedes/data

2.2. Data Preprocessing

Preprocessing and data analysis are the initial stages that must be done before starting to use and train the model [21]. Data *preprocessing* is done to prepare the raw data into a format that is ready to be used in the prediction model[22]. This process starts with identifying numerical features and calculating their correlation to price. Features that have a correlation above 0.50 or below -0.50 are considered significant and selected for further analysis. Other important features, such as year of production and mileage, are also included even though they are not numeric. After that, missing value checking and feature normalization were performed to ensure the dataset was ready for use. The process continued by converting categorical features to numeric using *one-hot encoding*, so that features such as vehicle model, transmission, and fuel can be understood by machine learning algorithms. Next, the dataset was split into features (independent variables) and targets (vehicle price as the dependent variable), and then features were normalized with *StandardScaler* to equalize the scale. The data is then divided into 80% training data and 20% test data for model accuracy evaluation. This stage ensures the data is ready to be used in *Machine Learning*.

2.3. Data Visualization

This process displays *scatter plots* to see the relationship between variables and histograms for the distribution of individual variables, with the aim of understanding patterns or correlations between variables in the dataset. In addition, a *jointplot* with *Kernel Density Estimate* (KDE) was used to display the data density of two variables at once. *The jointplot* was created to see

relationship between price and mileage, year of production, and engineSize. This visualization helps to understand the price distribution based on these factors and identify possible patterns.

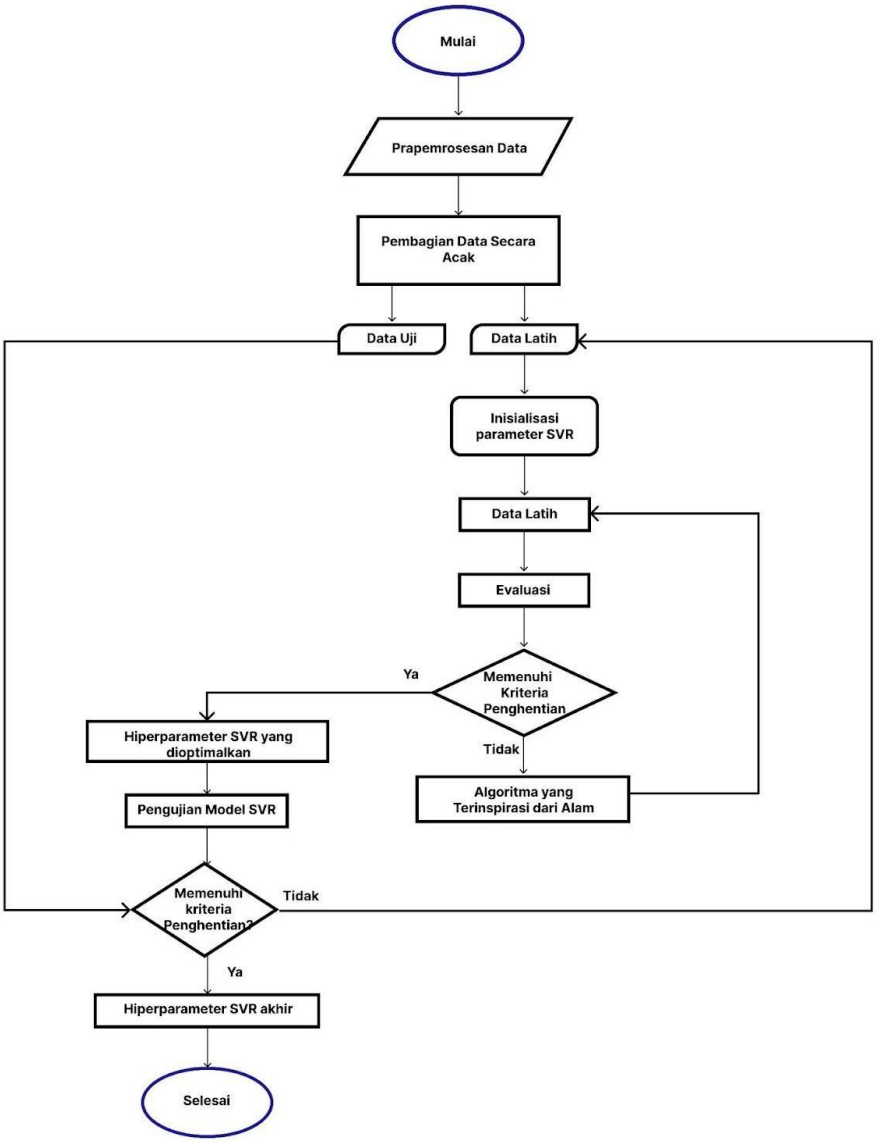
2.4. Modeling

2.4.1. Support Vector Machine Algorithm

Support Vector Machine (SVM) serves to train models in Machine Learning, especially in cases with a limited amount of data. This method was developed by a scientist named Vapnik and his team at Bell Laboratory in 1955. *Support Vector Regression* (SVR) uses the principle of *Support Vector Machine* (SVM) to solve regression problems... The fundamental difference between SVM and SVR lies in the application goal; SVM is focused on identifying the optimal hyperplane to separate two classes of objects, while SVR aims to find a function that acts as a hyperplane representing the regression line based on the entire input data [23]. In this process, SVR tries to minimize the error (ε) so that the value can be as low as possible [24].

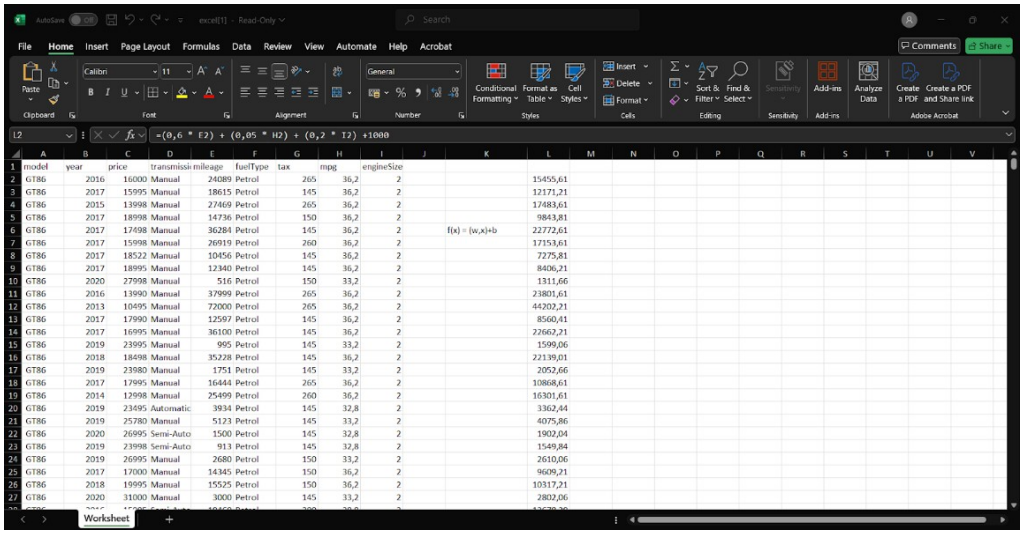
$$f(x) = \mathbf{w} \cdot (\sum_{i=1}^{n} . \mathbf{X}_i) + b \quad (1)$$

Equation (1) is used as the basis for calculating the output value in *Support Vector Regression*. The objective function in *Support Vector Regression* (SVR) is designed to achieve a balance between model complexity and fault tolerance. Parameters
C which serves as a regularization factor, which determines how much the model can "violate" the allowed margin or deviation, with the aim of maintaining equilibrium
between *overfitting* and generalization. SVR aims to find the optimal hyperplane that can accurately predict the output value. This process involves minimizing the objective function while adhering to various constraints, using *quadratic* optimization or *quadratic programming* methods in order to achieve the most optimal solution.



Support Vector Regression Flowchart

The flowchart in **Figure 2.** explains how the *Support Vector Regression* algorithm works. This figure shows the process of training and evaluating the *Support Vector Regression* (SVR) model. The process begins with the data preparation stage, which includes data preprocessing and random division into test set and training set. Next, the initial SVR parameters are initialized. The SVR model is then trained using the training set. The training results are evaluated, and if the stopping criteria are met, the optimal SVR parameters are obtained. Otherwise, the *Nature Inspired Algorithm* is used to update the SVR parameters.



Excel Implementation of SVR Formula

The excel calculation in **Figure 3.** shows a simple implementation used to calculate a value based on a combination of three main variables that have certain weights. The weights and biases have values that have been determined during the coding process, so the weight variations in excel are just an example of the SVR formula implementation. The first variable (E2) has the largest contribution of 60%, the second variable (H2) contributes 5%, and the third variable (I2) contributes 20% to the final result. In addition, there is an additional constant of 1000 that serves as a baseline adjustment or bias value. This model can be applied in various contexts, such as estimating used car prices based on mileage, vehicle tax, and engine size. This combination of variable weights and constants provides flexibility in customizing the results as needed.

Table 1. SVR Pseudocode

```
x: mileage
y: tax
z: engineSize

function calculateValue(x, y, z):
    Weight for each variable weight_x = 0.6
    weight_y= 0.05
    weight_z = 0.2
    constant = 1000

    Calculate the final score
    Output = (weight_x * x) + (weight_y * y) + (weight_z * z) +
constant

    Return result
    return Output
```

The *pseudocode* in **Table 1** illustrates the workflow to optimize the *Support Vector Regression* (SVR) model using a nature-based approach. The process begins with the data preparation stage, including preprocessing and randomly dividing the data into a training set and a testing set. Next, the initial SVR parameters are initialized.

Then, the SVR model is trained using the training set and evaluated for performance. If the stopping criteria are not met, a nature-based algorithm will be used to update the SVR parameters.

2.4.2. Simulated Annealing Algorithm

The *Simulated Annealing* (SA) algorithm is inspired by the physical process of annealing in metallurgy, where materials are heated to high temperatures and then cooled slowly to achieve a more stable structure [25], [26]. In optimization, this principle is used to find optimal solutions in problems with large and complex solution spaces. The SA algorithm starts by randomly selecting an initial solution, then making small changes t o that solution to produce a new solution. If the new solution is better, it is adopted. However, if it is worse, it may still be accepted based on a temperature-dependent probability or "temperature" in the algorithm. This temperature decreases gradually over time, so initially bad solutions may be accepted to allow the algorithm to explore the solution space more widely and avoid local solution traps. However, over time, acceptance of worse solutions becomes less frequent as the temperature decreases, which forces the algorithm to search for the best solution around the optimal point. This method is effective for complex non-linear problems, as it is able to escape the suboptimal solution trap and approach the global optimal solution.

$$P(X) = e^{-\frac{\Delta F}{K}} \tag{2}$$

Equation 3 determines how likely it is that a new solution will be accepted even if it is of worse quality than the current solution.
Description in the equation

- P(X) = Probability of accepting the new solution.
- ΔF = Difference between the energy of the new solution and the energy of the current solution.
- K = The current temperature (control parameter) in the *Simulated Annealing* process.

The temperature K, which is initially high, will decrease during the iterations of the algorithm. When K is high, the value in the above formula will be larger, giving a greater chance of accepting a bad solution, allowing the algorithm to exit the local solution. However, as K decreases, this probability also decreases, making the algorithm more selective in accepting new solutions, so that in the final stage only truly better solutions are likely to be accepted. This process maintains a balance between initial exploration (broad solution search) and final exploitation (refinement towards the optimal solution).

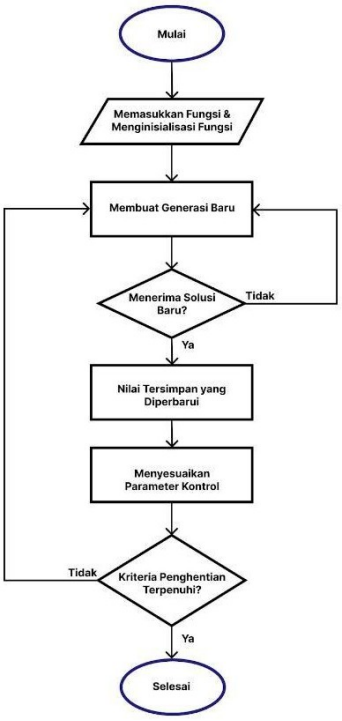


Figure 4. Simulated Annealing Flowchart

The flowchart in **Figure 4** illustrates an iterative process to optimize a solution. The process begins by defining input functions and initial functions, then generating new solutions. The new solution is then evaluated to determine whether it is acceptable or not. If accepted, the stored values are updated. Otherwise, the process will return to the new solution generation stage. Furthermore, the temperature will be adjusted, which may be related to the algorithm's stopping criteria. If the stopping criteria are met, the process will end. Otherwise, the process will return to the generation of new solutions.

Table 2. SA Pseudocode

```
X = number of movements
attempted; K = control parameter;
for x=1 to N {

    Create random movements, for example, moving
particles;
    Energy change evaluation, F;
    if (F < 0) {
        /* downward movement: accept /
        accept this movement, and update the configuration;
    }
    else{
        / climbing movement: accept possible /
     accept with probability P(X) = e-F/K; update
     configuration if accepted;
    }
}/*end for loop*/
```

The *pseudocode* in **Table 2.** illustrates the optimization process of the *Support Vector Regression (*SVR) model using the *Simulated Annealing* algorithm. Starting with data pre-processing and division of the dataset into training and testing data, the initial SVR parameters are initialized. Then, the model is trained and evaluated iteratively, with a stopping criterion that is continuously checked. If the criteria are not met, new parameters are randomly generated and evaluated, and acceptance of the new solution is decided based on performance and probability. After the iteration is completed, the optimized SVR model is tested using the test data, and the final result is retrieved if it meets the optimization criteria.

3. RESULTS AND DISCUSSION

3.1. *Dataset*

This dataset titled "100,000 UK Used Car Data Set" is taken from Kaggle and will be used as the main material in the research to train a machine learning model that can predict used car prices. The dataset includes a total of 13 CSV files containing various information about used cars in the UK, one of which is the file "toyota.csv". The "toyota.csv" file focuses on the information of Toyota brand used cars, with 9 attribute columns containing various features that affect the price of the car, such as year of production, mileage, engine size, fuel type, and transmission. The dataset consists of 6,738 data entries, which provides a wide range of information on the different variants of Toyota cars in the used car market.

	model	year	price	transmission	mileage	fuelType	tax	mpg	engineSize
0	GT86	2016	16000	Manual	24089	Petrol	265	36.2	2.0
1	GT86	2017	15995	Manual	18615	Petrol	145	36.2	2.0
2	GT86	2015	13998	Manual	27469	Petrol	265	36.2	2.0
3	GT86	2017	18998	Manual	14736	Petrol	150	36.2	2.0
4	GT86	2017	17498	Manual	36284	Petrol	145	36.2	2.0

Figure 5. Displaying 5 Initial Data

T h e dataset "toyota.csv" as seen in Figure 5. has 9 main features used to predict the price of used cars. The following is an explanation of each feature:

- Model: Toyota car model.
- year: Year of car production.
- Price: The price of the used car in GBP (Pounds Sterling), which is the target variable for prediction.
- Transmission: The type of transmission of the car (for example, manual or automatic).
- mileage: The distance traveled by the car in miles.
- fuelType: The type of fuel the car uses (e.g., gasoline, diesel).
- tax: Annual vehicle tax in GBP.
- mpg: The fuel efficiency of a car, measured in miles per gallon.
- engineSize: The size of the car engine in liters.

3.2. *Data Exploration*

Exploratory Data Analysis (EDA) focuses on exploring, understanding, and cleaning the data as a first step before moving on to statistical analysis or data mining.

Modeling. This step is done to recognize patterns, study the characteristics of the data, detect outliers, and understand the relationship between the variables.

3.2.1. *Data Form and Information*

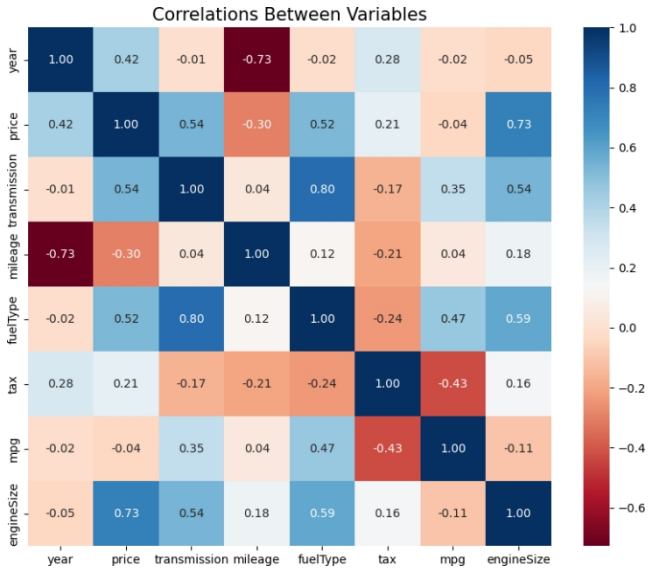
The first step in this stage involves displaying information about the size of the dataset, such as the number of rows and columns, to give an initial idea of the scale of the data to be managed. Next, using df.info() provides an in-depth summary of the dataset structure, such as the number of non-null values in each column, the data type of each column (e.g. numeric or categorical), and an estimate of the memory used. This information is very useful for detecting potential problems such as *missing values* or data type mismatches. Finally, it provides descriptive statistics such as mean, standard deviation, minimum and maximum values, and quartiles for all numeric columns in the dataset.

3.2.2. *Heatmap Correlation*

This correlation aims to measure how strong the linear relationship between each feature and price is. The correlation results are visualized in the form of a heatmap to provide a clear picture of the features that have a significant relationship. In this research, heatmap correlation is used to show the relationship between various variables in the car dataset. Correlation is measured in the range of -1 to 1, where positive values indicate a unidirectional relationship (when one variable increases, the other variable also tends to increase), while negative values indicate an opposite relationship (when one variable increases, the other variable tends to decrease). In this heatmap, dark blue indicates a strong positive correlation, while dark red indicates a strong negative correlation, and lighter colors in between indicate a weaker correlation or no correlation.

3.3. *Data Preprocessing*

Preprocessing begins by identifying numerical features in the dataset based on the correlations found in the heat map. Features that have a correlation of more than
0.50 or less than -0.50 against the price column were considered to have a significant influence and were selected for further analysis. In addition, some additional relevant features, such as year of production and mileage, were also included in the analysis even though they are not numerical features. Next, missing values in the selected columns were checked. This check ensures that the dataset is clean and ready to use without any missing values. After that, significant numerical features were normalized to equalize the scale between features, so that the model can work more optimally. This stage results in the formation of a subset of features that have the most influence on vehicle prices, which are then used as inputs in the prediction model.



Heatmap Correlation

The correlation *heatmap* shown in Figure 6. shows the identified patterns, where the negative correlation between 'year' and 'mileage' (-0.73) is quite strong, meaning that newer cars tend to have lower mileage. The significant positive correlation between 'price' and 'engineSize' (0.73) indicates that cars with larger engines tend to be more expensive. Engine size is usually related to car performance, so it is not surprising that cars with larger engines are sold at higher prices. The positive correlation between 'transmission' and 'fuelType' (0.80) indicates a strong relationship between transmission type and fuel type. This could mean that cars with a specific transmission type tend to use a specific fuel type. The variable 'tax' has a moderate negative correlation with 'mpg' (-0.43). This suggests that cars with higher fuel efficiency (greater mpg) tend to have lower taxes. This makes sense as more fuel-efficient cars often get lower tax incentives in many countries.

The preprocessing stage is followed by converting categorical features into numerical form using the *one-hot encoding* method. This is done to ensure that categorical features, such as vehicle model, transmission type, and fuel type, can be understood by *machine learning* algorithms. After that, the dataset is separated into two parts, namely features (independent variables) and targets (dependent variables) where features consist of all columns other than the price column, while the target is the vehicle price to be predicted. The next step is feature normalization using the *StandardScaler* method.

$$Z = \frac{x - \mu}{\sigma} \tag{3}$$

Description:
Z = Standardized value of each feature X μ = *Mean* (X)
σ = Standard Devization

Normalization is done to ensure that all features have a balanced scale, which can improve the performance of the model. Finally, the data is divided into two parts with a proportion of 80% for training data and 20% for test data. This allows the model to be trained and tested on different data for more accurate performance evaluation. All these steps ensure the data is ready to be used in *Machine Learning.*

3.4. *Data Visualization*

This process displays the relationship between variables in the form of *scatter plots* for each pair of variables, as well as the distribution of individual variables in the form of histograms. This visualization aims to get an initial idea of how the variables in the dataset relate to each other and whether there is a significant pattern or correlation. displays the distribution of two variables simultaneously and adds a *Kernel Density Estimate* (KDE) to see the density of the data in two dimensions. The first *jointplot* displays the relationship between mileage and price. The second and third jointplots, respectively, depict the relationship between year and engineSize and *price*. This visualization helps to understand the distribution of price based on factors such as mileage, year of production, and engine size, and whether certain patterns can be identified.

The visualization in this study makes use of contour plots to help identify relationships and patterns between input variables and price variables, which provides deeper *insight into* the factors that influence car prices. *Contour plot* is a visualization used to show the relationship between two continuous variables, where contour lines depict data density and distribution patterns along the two variable axes. *Contour plots* are very useful in regression analysis and prediction as they provide a visual representation of the influence of each feature on price, aiding in the selection of features that have a significant impact in the model. In this study, each contour plot shows the interaction of two main features, such as 'mileage', 'year', and 'engineSize', with the car price feature or 'price', making it possible to observe data distribution patterns as well as areas of high concentration.

3.4.1. *Distribution of Car Price (price) to (mileage)*

The plot distribution graph shows that most cars have low mileage, with the distribution rapidly decreasing after reaching around 50,000 kilometers. On the right side, the price distribution graph shows that most of the cars analyzed are in the price range below $20,000. Overall, these plots show that cars with low mileage tend to have higher prices, and there is a negative relationship between mileage and price, where prices tend to decrease as mileage increases.

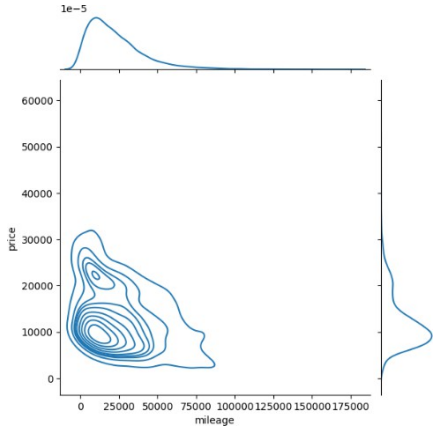


Figure 7. Distribution chart of mileage against price

The *contour plot* in **Figure 7** displays the relationship between two variables, mileage on the x-axis and price on the y-axis. The contour lines indicate the density of the data, where areas with tighter lines indicate a higher concentration of data. In this figure, we can see that most cars have a mileage below 50,000 kilometers, with prices ranging from $5,000 to $20,000. Outside of this range, cars with higher mileage tend to have lower prices, but some cars are still seen to have prices above
$20,000 even if the mileage is greater than 50,000Km.

3.4.2. *Car Price Distribution (price) Against (year)*

The relationship between car year on the x-axis and car price on the y-axis is also shown in Figure 8 in the form of a Contour plot visualization. The contour lines show the density of the data distribution, with denser areas indicating more data within that range.

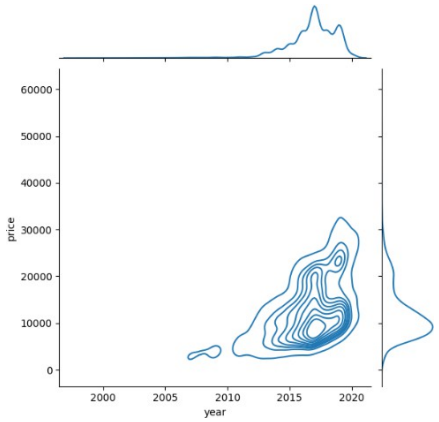


Figure 8. Distribution chart of year against price

On the x-axis, it can be seen that most of the analyzed cars were manufactured between 2010 and 2020. The prices of cars in this year range from $5,000 to over $30,000. Most cars with newer production years (2015 and above) have higher prices, with peak prices ranging from around $20,000 to $30,000. On the right-hand side, the price distribution chart shows that most cars

below $20,000, but some cars reach higher prices of over $60,000. The distribution graph at the top of the plot shows that the majority of the cars analyzed were produced after 2010, with few cars produced before that year. In addition, there is a spike in car production between 2015 and 2020, reflecting newer cars. Overall, this plot shows that newer cars tend to have higher prices, and most of the data is centered on cars produced after 2010, with prices ranging between $10,000 to $30,000.

3.4.3. *Distribution of Car Price (price) against (engineSize)*

A visualization of the relationship between the *engineSize* and *price* variables in the regression analysis of used car prices is shown in Figure 1. This contour graph depicts the distribution of data as well as the density of points in a two-dimensional space, showing the pattern of price concentration by engine *size*. It appears that prices tend to be distributed around a certain engine size range, indicating a trend that can be utilized in the prediction model.

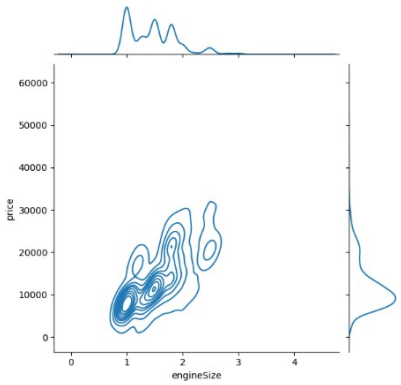


Figure 9. EngineSize distribution graph against price

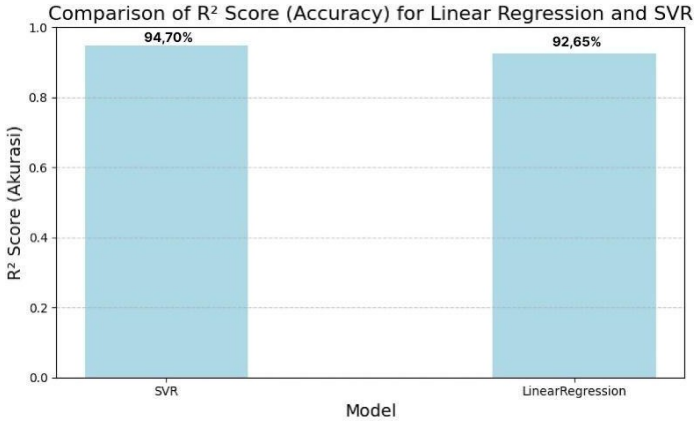
A *contour plot* depicting the relationship between two variables, engine size and price, is shown in **Figure 9**. On the x-axis, there is the engine size variable which ranges from 0 to about 4 liters, while the y-axis shows the price of the car which ranges from 0 to over 60,000. This contour map shows the concentration of data in the form of lines that form a pattern, with denser areas of lines indicating a higher concentration of data. Based on the figure, it can be seen that most cars with engine sizes between 1 and 2 liters are in the price range between $5,000 and $20,000. There are some cars that cost more than $20,000, but this seems to be a rarer case. On the top and right side of the plot, there are data distribution graphs. The top graph shows the engine size distribution, where most cars have engine sizes below 2 liters. The right-hand graph shows the price distribution, where most cars are priced below $20,000. Some other features such as car model, transmission type, and fuel type have relatively little influence on the price prediction. This means that, while they may be relevant in certain contexts, they do not play a major role in determining the final value of a used car.

The visualization provides information on how each variable affects the price, which can be used as a reference in choosing which features to prioritize or ignore in the prediction model.

3.5. *Algorithm*

3.5.1. *Support Vector Regression*

The *Support Vector Regression* (SVR) model uses a regularization parameter (C = 3000). This (C) value determines how much penalty is given to the prediction error, where the higher the value, the smaller the tolerance margin allowed for error. The model is trained using the training data (*X_train and y_train*) to understand the relationship pattern between the independent features (predictors) and the target variable, which is the car price. Once the training process is complete, the model is then used to predict the price on the test data (*X_test*), resulting in predictions referred to as predictions. To evaluate the performance of the model, several metrics are used. In this study, we use the linear regression algorithm as a comparison algorithm.



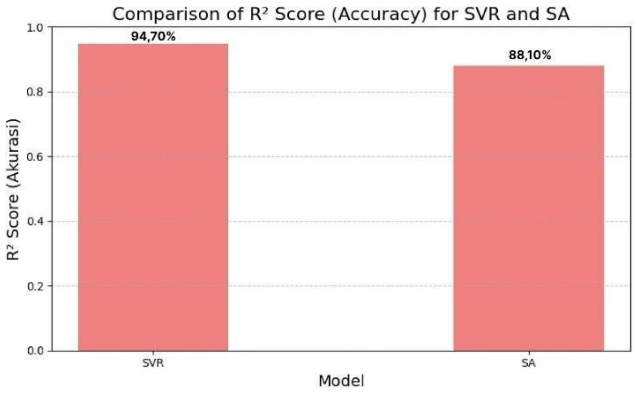
Comparison of SVR and Linear Regression Results

The results of the comparison of the accuracy of SVR and *Linear Regression* in **Figure 10.** SVR recorded a greater R² Score value compared to *Linear Regression*, namely 0.9470, while *Linear Regression* got 0.9265. This shows that the SVR model is superior to *Linear Regression*. SVR was able to explain 94.70% of the variability in the data. A higher R^2 value indicates that the model has a better ability to model the relationship between the independent variable and the dependent variable, making SVR a more efficient model in explaining data variation.

3.5.2. *Support Vector Regression Optimization with Simulated Annealing*

Support Vector Regression (SVR) modeling optimized with *Simulated Annealing* (SA) starts by defining an objective function. This function aims to minimize the negative *Root Mean Squared Error* (RMSE) value, which will help find the best combination of hyperparameters C and epsilon in SVR. In this function, first the parameters C and epsilon are taken from the simulation results. C affects the strength of the model's regulation of the data, while epsilon determines the margin at which the error

prediction is still considered good enough. The SVR model was initialized with the C and epsilon values generated at each SA iteration, then *cross-validated* three times to calculate the average negative RMSE value. Furthermore, SA was used as an optimization technique to find the best combination of C and epsilon based on predefined constraints (C between 1 and 100,000, and epsilon between 0.0001 and 1).



Comparison of SVR and SA Results

The results of the comparison of the accuracy of SVR and *Simulated Annealing* in **Figure 11.** SVR recorded a greater R^2 Score value compared to Simulated *Annealing*, namely 0.9470, while *Simulated Annealing* got 0.8810. This shows that the SVR model is superior to *Simulated Annealing*. SVR was able to explain 94.70% of the variability in the data. A higher R^2 value indicates that the model has a better ability to model the relationship between the independent variable and the dependent variable, making SVR a more efficient model in explaining data variation.

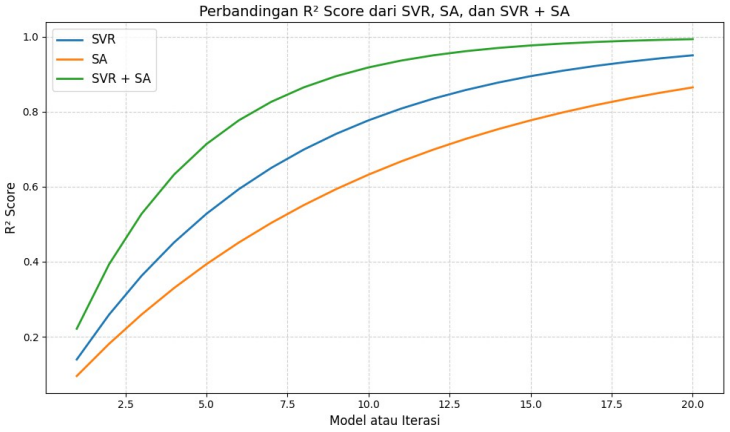


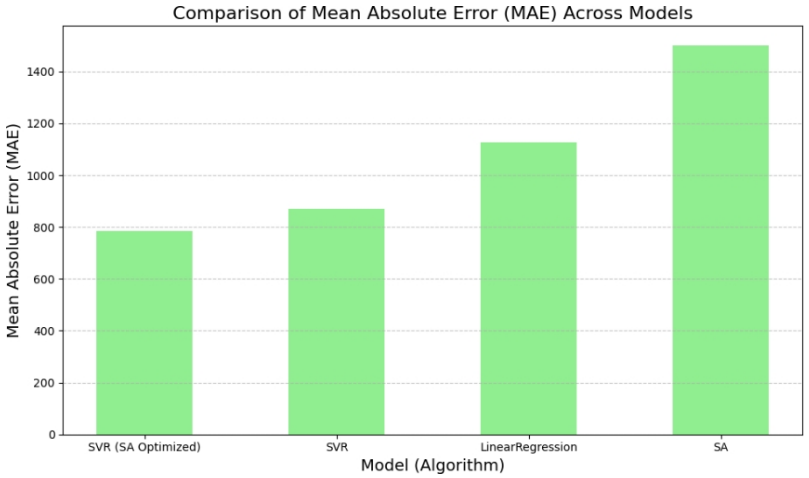
Figure 12. Comparison Chart of R^2 Score between SVR, SA, and SVR with SA

This optimization process sought the best solution by testing various combinations of C and epsilon that resulted in the lowest RMSE, utilizing SA's ability to effectively explore various possible parameter combinations. After this process, the best values for C and epsilon were found, which were approximately 786.10 for MAE, 1,467,156.40 for MSE, 1211.26 for RMSE, and R^2 of 96.55%. The comparison results between SVR, SA, as well as SVR (SA Optimized) are shown in **Figure 12.**

after being optimized using *simulated annealing*, the accuracy of SVR increased to 96.55%.

3.5.3. *Comparison Results of MAE, MSE, and RMSE Metrics*

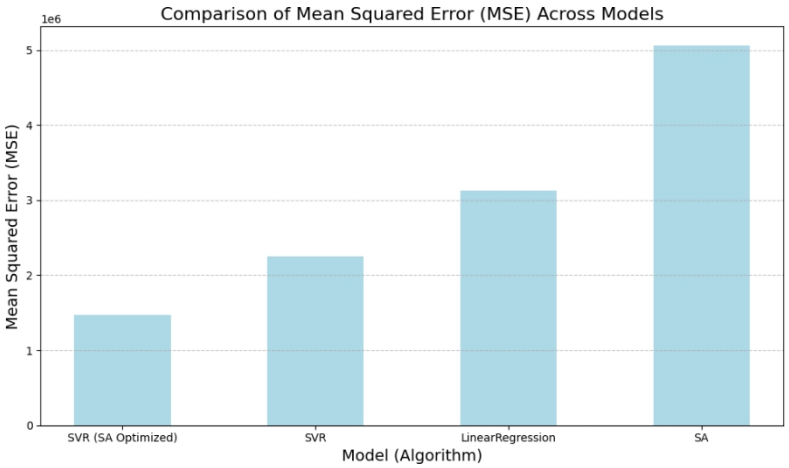
In evaluating regression models, the use of various error metrics, such as *Mean Absolute* Error (MAE), *Mean Squared Error* (MSE), *Root Mean Squared Error* (RMSE) is essential to assess the overall performance and accuracy of the model. **MAE** measures the average absolute error between the prediction and the actual value (predicted used car price and real used car price), giving an idea of how far the model prediction is from the actual value in the same unit as the target, thus helping in assessing the accuracy of the prediction in general. In MAE, the lower the value, the better the model performs as it indicates that the average difference between the predicted and actual values is smaller, indicating a more accurate prediction. **MSE** calculates the average square of the error, putting more emphasis on larger errors, which makes this metric useful for identifying models that are able to minimize extreme errors. In MSE, the lower the value, the better the model is at reducing large or extreme errors, as MSE gives higher weight to large errors by squaring them. **RMSE**, as the root of MSE, returns the error to the same units as the target and is often used to understand how large the overall model error is. RMSE is also better the lower it is, as it returns the error to the original scale of the target variable and shows how much the average prediction deviates from the actual value, which is important for keeping the error minimal. This research uses these various error metrics in order to find out which model has more optimal accuracy in predicting used car prices.



Comparison of MAE results

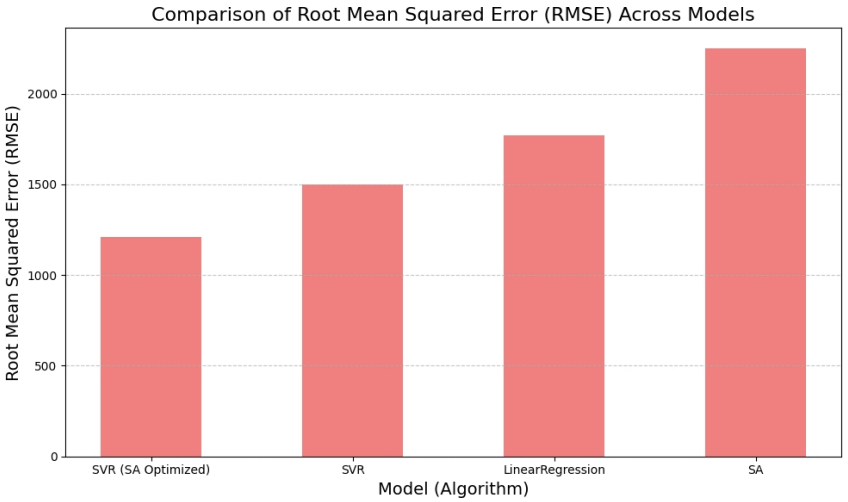
A comparison of the MAE evaluation of each model is shown in **Figure 13** to determine which model is most accurate in predicting the results. The *Support Vector Regression* (SVR) model optimized with *Simulated Annealing* (SA) has the lowest MAE value of **786.10**, showing the best performance with the smallest average absolute error. SVR without optimization resulted in an MAE **of 868.84**, which is still better than *Linear Regression* with the highest MAE of **1128.20**. On the other hand, the Simulated Annealing (SA) model as a standalone method had an MAE of **1472.56**,

shows that although SA is effective as an optimization technique, its performance as an independent model is less competitive than others. From these results, it can be concluded that the smaller the MAE value, the more accurate the prediction produced by the model. Therefore, SA-optimized SVR excels in terms of accuracy compared to other models.



Comparison of MSE results

Comparison of the MSE evaluation results in **Figure 13.** again shows that the SVR model with SA has the lowest MSE value of **1,467,156.40**, indicating that this model has the smallest squared deviation from the actual data.SVR without optimization produces an MSE value of **2,253,058.92**, which is higher than SVR with optimization. *The Linear Regression* model had an MSE value of **3,126,829.22**, showing worse performance than both SVR models. Meanwhile, *Simulated Annealing* (SA) as a standalone model has the highest MSE value of **5,003,126.45**, indicating that this model is less competitive when used without combination or optimization on other models.From these results, it can be concluded that the smaller the MSE value, the smaller t h e prediction error rate of a model. Therefore, SA-optimized SVR provides the most precise results compared to other models.



Comparison of RMSE results

Furthermore, the model performance evaluation is continued with *Root Mean Squared Error* (RMSE) analysis. RMSE is used to measure how large the average prediction error is in the same unit as the target variable. Figure 16 shows a comparison of *Root Mean Squared Error* (RMSE) values for various models. RMSE gives an idea of the prediction error in the same units as the target variable, so the smaller the value, the better the model performance. The *Support Vector Regression* (SVR) model optimized with *Simulated Annealing* (SA) again showed the best performance with the lowest RMSE value of **1211.26**, indicating the smallest and most consistent prediction error. SVR without optimization has an RMSE value of **1501.02**, which is better than the *Linear Regression* model with an RMSE of **1768.28**. However, the *Simulated Annealing* (SA) model as an independent method had the highest RMSE value of **2237.43**, indicating less than optimal performance when used as an independent model. Overall, a smaller RMSE value indicates that the model has a prediction error that is closer to the actual data. Therefore, SA-optimized SVR is the most precise model in prediction compared to other models.

4. CONCLUSIONS

From the research that has been done, it can be concluded that the most significant improvement is seen in the SVR model optimized with *Simulated Annealing*. This model produces a small MAE of 786.10, MSE of 1,467,156.40, RMSE of 1211.26, and a very high R^2 score of 0.9655. With this score, the model was able to explain 96.55% of the variation in the data and provide predictions with much smaller errors than the previous two models. In conclusion, the use of *Simulated Annealing* as an optimization method in the SVR model proved to provide the best performance, making it the most accurate choice for used car price prediction in this experiment...

5. ADVICE

For future research, it is recommended that additional features such as mileage, fuel efficiency, and maintenance history be investigated to improve prediction accuracy. Comparative studies with other optimization algorithms such as *Genetic Algorithm* or *Particle Swarm Optimization* can also be done to determine the best method, because the one we use *Simulated Annealing* is quite time-consuming to model.

ACKNOWLEDGMENTS

With all h u m i l i t y, we would like to express our deepest gratitude to all those who have supported and made valuable contributions in the completion of this research. My special thanks go to the supervisor and colleagues who have provided guidance, input, and motivation during the research process.

LITERATURE

[1] J. Huang *et al.*, "A Latent Factor-Based Bayesian Neural Networks Model in Cloud Platform for Used Car Price Prediction," *IEEE Trans Eng Manag*, vol. 71, pp. 12487-12497, 2024, doi: 10.1109/TEM.2023.3270301.

[2] M. Aji Saputra and U. Hayati, "ESTIMATION OF TOYOTA YARIS REVENUE PRICE USING LINear REGRESSION ALGORITHM," 2024. [Online]. Available: https://www.researchgate.net/figure/DataMi

[3] M. Hankar, M. Birjali, and A. Beni-Hssane, "Used Car Price Prediction using Machine Learning: A Case Study," in *11th International Symposium on Signal, Image, Video and Communications, ISIVC 2022 - Conference Proceedings*, Institute of Electrical and Electronics Engineers Inc. doi: 10.1109/ISIVC54825.2022.9800719.

[4] Y. Li, Y. Li, and Y. Liu, "Research on used car price prediction based on random forest and LightGBM," in *2022 IEEE 2nd International Conference on Data Science and Computer Application, ICDSCA 2022*, Institute of Electrical and Electronics Engineers Inc, 2022, pp. 539-543. doi: 10.1109/ICDSCA56264.2022.9988116.

[5] A. Kumar, D. Singh, K. Chauhan, and M. S. Guru Prasad, "An Effective Pre-Owned Car Price Prediction Based on Multi Linear Regression Technique," in *2023 International Conference on Computer Science and Emerging Technologies, CSET 2023*, Institute of Electrical and Electronics Engineers Inc. doi: 10.1109/CSET58993.2023.10346830.

[6] G. Najla, A. #1, and D. Fitrianah, "Application of Linear Regression Method for Property Sales Prediction at PT XYZ," *Journal of Telematics*, vol. 14, no. 2.

[7] S. Han, J. Qu, J. Song, and Z. Liu, "Second-hand Car Price Prediction Based on a Mixed-Weighted Regression Model," in *2022 7th International Conference on Big Data Analytics, ICBDA 2022*, Institute of Electrical and Electronics Engineers Inc, 2022, pp. 90-95. doi: 10.1109/ICBDA55095.2022.9760371.

[8] W. Wilianto, Y. Yuliana, A. Suwandhi, J. Jimmy, and J. Putra, "Application of AI in Determining Used Car Prices Based on Year of Assembly," *Journal of Minfo Polgan*, vol. 13, no. 1, pp. 550-560, Jun. 2024, doi: 10.33395/jmp.v13i1.13728.

[9] M. Ahmad *et al.*, "Car Price Prediction using Machine Learning," in *2024 IEEE 9th International Conference for Convergence in Technology, I2CT 2024*, Institute of Electrical and Electronics Engineers Inc. doi: 10.1109/I2CT61223.2024.10544124.

[10] J. Yang *et al.*, "Metaverse: Design of the Car Price Prediction Model Through a Machine-learning Approach," in *Proceedings - 2023 IEEE International Conference on Metaverse Computing, Networking and Applications, MetaCom 2023*, Institute of Electrical and Electronics Engineers Inc, 2023, pp. 734-737. doi: 10.1109/MetaCom57706.2023.00139.

[11] F. Wang, X. Zhang, and Q. Wang, "Prediction of Used Car Price Based on Supervised Learning Algorithm," in *Proceedings - 2021 International Conference on Networking, Communications and Information Technology, NetCIT* 2021, Institute of Electrical and Electronics Engineers Inc, 2021, pp. 143-147. doi: 10.1109/NetCIT54147.2021.00036.

[12] J. Varshitha, K. Jahnavi, and C. Lakshmi, "Prediction Of Used Car Prices Using Artificial Neural Networks And Machine Learning," in *2022 International Conference on Computer Communication and Informatics, ICCCI 2022*, Institute of Electrical and Electronics Engineers Inc. doi: 10.1109/ICCCI54379.2022.9740817.

[13] J. S. Jhala and D. Anand, "Comparative Analysis of Supervised Learning Algorithms for Valuating Used Car Prices," in *2023 International Conference on Advancement in Computation and Computer Technologies, InCACCT 2023*, Institute of Electrical and Electronics Engineers Inc, 2023, pp. 265-270. doi: 10.1109/InCACCT57535.2023.10141827.

[14] H. Drucker-, C. J. C. Burges, L. Kaufman, A. Smola-, and V. Vapoik, "Support Vector Regression Machines."

[15] A. W. Ishlah, S. Sudarno, and P. Kartikasari, "IMPLEMENTATION OF GRIDSEARCHCV ON SUPPORT VECTOR REGRESSION (SVR) FOR PRICE FORECASTING
SHARE," *Gaussian Journal*, vol. 12, no. 2, pp. 276-286, Jul. 2023, doi: 10.14710/j.gauss.12.2.276-286.

[16] R. Wahyudi, S. Annas, and Z. Rais, "SUPPORT VECTOR REGRESSION (SVR) ANALYSIS FOR FORMULATING AIR QUALITY INDEX IN CITY MAKASSAR," *VARIANCE: Journal of Statistics and Its Application on Teaching and Research*, vol. 5, no. 3, pp. 104-117, 2023, doi: 10.35580/variansiunm107.

[17] S. Saadah, F. Zahra, and H. Haifa, "Support Vector Regression (SVR) To Predict Crude Oil Palm Price in Indonesia and Exchange Rate of EUR/USD." [Online]. Available: http://jcosine.if.unram.ac.id/

[18] Y. Lin, R. Wu, Y. Yue, and Q. Liao, "Forecasting gold price using a novel hybrid model with MEEMD-convLSTM CRediT authorship contribution statement." [Online].
Available: https://ssrn.com/abstract=4591257

[19] S. S. Rocha, C. S. Pitombo, L. H. M. Costa, and S. de F. Marques, "Applying optimization algorithms for spatial estimation of travel demand variables," Jun. 01, 2021, *Elsevier Ltd.* doi: 10.1016/j.trip.2021.100369.

[20] G. Eko Noviardianto, M. Novel, and M. Broto Legowo, "Use of Simulated Annealing Method for Access Point Positioning Optimization in WI-FI Network," 2019.

[21] S. K. Satapathy, R. Vala, and S. Virpariya, "An Automated Car Price Prediction System Using Effective Machine Learning Techniques," in *Proceedings of International Conference on Computational Intelligence and Sustainable Engineering Solution, CISES 2022*, Institute of Electrical and Electronics Engineers Inc, 2022, pp. 402-408. doi: 10.1109/CISES54857.2022.9844350.

[22] W. Andriani, Gunawan, and A. E. Prayoga, "PREDICTION OF GOLD VALUE USING LINEAR REGRESSION ALGORITMA," *Scientific Journal of Computer Informatics*, vol. 28, no. 1, pp. 27-35, 2023, doi: 10.35760/ik.2023.v28i1.8096.

[23] H. Zhong, J. Wang, H. Jia, Y. Mu, and S. Lv, "Vector field-based support vector regression for building energy consumption prediction," *Appl Energy*, vol. 242, pp. 403-414, May 2019, doi: 10.1016/j.apenergy.2019.03.078.

[24] A. Yaqin, M. Rahardi, F. F. Abdulloh, Kusnawi, S. Budiprayitno, and S. Fatonah, "The Prediction of COVID-19 Pandemic Situation in Indonesia Using SVR and SIR Algorithm," in *Proceeding - 6th International Conference on Information Technology, Information Systems and Electrical Engineering: Applying Data Sciences and Artificial Intelligence Technologies for Environmental Sustainability, ICITISEE 2022*, Institute of Electrical and Electronics Engineers Inc., 2022, pp. 570-573. doi: 10.1109/ICITISEE57756.2022.10057813.

[25] B. Aylaj and S. Nouh, "Degeneration vs Classical of simulated annealing algorithm: performance analysis," in *Proceedings - 2022 5th International Conference on*

Advanced Communication Technologies and Networking, CommNet 2022, Institute of Electrical and Electronics Engineers Inc, 2022. doi: 10.1109/CommNet56067.2022.9993814.

[26] K. Liu, W. Sheng, and Y. Li, "Research on Reactive Power Optimization based on Adaptive Genetic Simulated Annealing Algorithm," Power System Technology, 2006.