# Machine Learning Optimization and Challenges in Used Car Price Prediction

*Yufan* Zheng

Santa Monica College, California, 90401, United States of America

**Abstract.** With the rapid expansion of the second-hand vehicle market, correctly forecasting car prices is essential for both researchers and industry experts. The paper initially reviews existing machine learning models and their performance in predicting luxury car prices, emphasizing both their strengths and limitations. To begin with, models like XGBoost and Random Forest excel at processing large-scale data and identifying complex feature patterns, thanks to their ability to use an ensemble of decision trees to reduce bias and variance. However, these models struggle to accurately capture the unique characteristics of luxury vehicles, such as brand reputation, rarity, and personalized configurations. Because these complex factors cannot be easily represented by simple numerical features, the result is often suboptimal predictions for high-value vehicle prices. The paper found that feature engineering could enhance model performance by introducing more representative attributes specific to luxury vehicles, such as brand reputation, rarity, and customization options. Additionally, stratified modeling, which segments data based on price tiers, may provide more accurate predictions by targeting different price levels, especially in the high-value vehicle segment. Despite these theoretical benefits, the paper acknowledges that while these strategies were discussed, they were not empirically tested in detail. Consequently, their practical effectiveness still requires further investigation.

## 1 Introduction

Recent growth in the used car market prompted attempts to accurately predict pricing. Accurate price prediction has become increasingly important in the used car market, and this demand is reflected across various industries. For instance, accurate price forecasting in the energy market is also highly valued to support better decision-making [1]. That being so these models work have their own problems due to the scale of data and market noise. For researchers and practitioners, the interesting question is how to use machine learning algorithms like XGBoost, Random Forests or Linear regression on data where vehicle attributes are treated as possible features that can help predict market values. In the used car market, vehicle lifetime and scrappage behavior also play a significant role in price fluctuations, adding complexity to model predictions [2]. These models have proven effective in predicting second-hand car prices by leveraging a range of numerical and

Corresponding author: zheng_yufan01@student.smc.edu

categorical features, including mileage, price, and brand [3]. However, despite these advancements, challenges remain, particularly in predicting prices for high-value vehicles. Traditional models often fail to account for the unique features of luxury cars, leading to skewed predictions in these segments [4].

Although machine learning models perform well in predicting the prices of regular vehicles, they face significant challenges when it comes to high-value luxury cars and other high-end goods. The unique characteristics of these products, such as brand reputation, rarity, and personalized configurations, make it difficult for traditional models to accurately capture their market value, resulting in biased predictions [5]. As a result, simpler models tend to miss crucial patterns in the data, resulting in inaccurate predictions for both low- and high-priced vehicles. More complex algorithms, such as Gradient Boosting and Random Forest, are necessary to address this issue by better capturing the intricate relationships between vehicle features and prices [6]. These advanced models are particularly effective in managing large datasets and mitigating the influence of outliers and anomalies, which are common in high-value segments.

In addition to improving model complexity, feature engineering is essential for enhancing prediction accuracy. High-value vehicles possess unique characteristics, such as brand reputation, rarity, and luxury configurations, that must be integrated into the model to improve its ability to predict their prices accurately. By incorporating these features, researchers can significantly reduce prediction errors for luxury cars. Furthermore, stratified modeling, which involves segmenting the data into different price tiers, can further enhance model accuracy by allowing for more targeted predictions within each price segment [7].

The purpose of this paper is to explore the effectiveness of machine learning models in predicting second-hand car prices, particularly focusing on the challenges associated with high-value luxury vehicles. To begin with, the paper will review existing models, highlighting their strengths and limitations when applied to the luxury car segment. Following this, the discussion will shift to advanced techniques such as feature engineering and stratified modeling, examining how these methods can enhance the accuracy of predictions. In addition, a case study will be presented to illustrate the practical application of these strategies, demonstrating their impact on improving price prediction for high-value vehicles.

## 2 Data and Methods

### 2.1 Data Source

The data in this article is a second-hand car price dataset. This dataset comes from the Kaggle website (https://www.kaggle.com/competitions/kaggle-1-second-hand-car-prices/data). This dataset contains attribute information for over 350,000 used cars, with nearly hundreds of thousands of records.Presented in the form of results from a pre-owned vehicle marketplace, this data includes information on hundreds of thousands of cars for sale. This dataset includes numerical and categorical features. The numerical features include attributes such as mileage and price, while the categorical features include brand, model, and transmission type. The combination of attribute diversity means that more factors can be examined in aggregate relative to car price.

The dataset is both expansive in its coverage, as well as rich with attributes that can be analyzed to interpret the value of used vehicle pricing for drivers. The preprocessor takes care of common problems such as: how to deal with missing data, creation of new more sophisticated features based on existing records and standardizin

g the data so that all attributes have a mean value of zero and a standard deviation equal to one. This process is necessary to increase the efficiency of models and i mportantly prevent potential biases.

### 2.2 Methods and Models

To analyze, as well as ascertain the price features of second-hand car prices, this paper uses a variety of machine learning algorithms. Seleted models are XGBoost. Random Forest, and Linear Regression These models have properties that make them suitable for different parts of the prediction task:

Extreme Gradient Boosting (XGBoost) is a very efficient gradient-boosting algorithm that works great with structured data. It does so by building an ensemble of decision trees, each itertively trying to correct the errors of the previous ones. The main thing in favor of the model: it is fast, efficient, and can manage overfitting with regularization techniques.

Random forest is made up of a series of decision trees generated by an algorithm and the output class depends on majority voting. Random Forest is a technique that by averaging over the results of multiple trees reduces variance, and in this way by extension improves generalization. Because the algorithm can handle high-dimensional data and extract important features.

Linear regression is a straightforward algorithm which can give people an easily interpretable relationship between the target variable and input attributes by fitting a linear equation. It serves as a foundation model that can simplify how each attribute directly affects car prices.

By utilizing ensemble models, the accuracy of different methods is controlled based on their complexity, interpretability, and efficiency. The churn prediction models are fine-tuned using RandomizedSearch CV, which efficiently discovers combinations of hyperparameters to maximize model performance.

### 2.3 Evaluation Metrics

The models are evaluated using Root Mean Squared Error (RMSE) as the main evaluation metric. It simply measures the average magnitude of the error in terms of predicted and actual values. It is the square root of the average of squared differences between real targets and predicted values. In a regression setup, RMSE would be a handy model performance metric since it gives interpretable context to prediction accuracy (as lower the better) RMSE is then useful because it tells us how closely the model actually predicts prices a pre-owned car and where prediction depart significantly from true values.

In addition, Mean Absolute Error (MAE) and R-squared are used to comprehensively analyze and evaluate the algorithm's performance. MAE is likewise a popular metric that measures the average absolute difference between predicted values and actual ones. Unlike RMSE, MAE is not very sensitive to outliers. This is beneficial when the paper wishes to measure only the average magnitude of the errors in predictions without any additional weight on large differences.

## 3 Detailed step-by-step plan.

Feature engineering was carried out to augment the predictability of the model. This was achieved by loading the data and selecting relevant attributes, such as extracting horsepower from the engine column as a floating-point feature and calculating car_age to preserve valuable price-determining information. During this process, careful attention was

given to data type conversions and handling missing values and outliers in the predictors, as they can reduce model efficiency[8]. Certain features that did not perform well on pre-API 21 devices were decoupled. The database was simplified to contain only relevant information, and older attributes were removed once new ones were extracted. However, it is crucial to confirm that the feature is not required by any downstream process before its removal.

### 3.1 Data Preprocessing

Data preprocessing is the transformation of raw data into a more suitable form for machine learning. It involves treating missing values, scaling numerical features, and encoding categorical variables. The task is to transform data so that it fits into a set of requirements, purpose being use this transformed Data for feed data Analysis/ Traditional Machine learning models by Making sure Data is consistent and Clean.

Preprocessor Construction:

This preprocessing pipeline is responsible for two key tasks: processing numerical attributes and management of other categorical attributes. For the numerical attribute mileage, horsepower, and car_age; the paper uses mean imputation to fill in missing values and then standardizes these columns using StandardScaler. That way all numeric attributes will have the same scale, which allows the model to be more stable.

Similarly, the paperimpute missing values of the categorical attributes (brand and model) with mode imputation, which fills any null value in categorical feature by its higher repeated category. Post imputation, the categorical attributes are one hot encoded that is they get converted into binary formats which can be suitable to feed in algorithms for training.

In order to get this right, the paper needs ColumnTransformer and PipeLines. This helps us integrate all the preprocessing steps into one keeping the data transformation consistent.

Mistake prevention: Procedurally, the disturbances with numerical and categorical preprocessing methods due to their particular nature must have discrepancies. For example, trying to make a categorical attribute standardized can result in incorrect model inputs. Another important thing is to properly set up ColumnTransformer so that all the attributes are transformed in the correct way.

### 3.2 Define and Tune Model

The model definition and tuning phase: this is selecting the right algorithm that fits best for the machine learning process/optimizing those parameters to get greater efficiency. The paper tries to compare different models — XGBoost, Random Forest, and Linear Regression about how well they do in predicting pre-owned vehicle prices with this project.

Parameter Tuning:

Parameter tuning is one of the major parts of an efficient model. The paper uses RandomizedSearchCV instead of full GridSearchCV to save some time without losing the benefits of a wide hyperparameter search. This method is better as it reduces the number of combinations in large search spaces which means, the paper can pick up the right set of parameters faster and more efficiently.

And the paper will tune learning rate and max three depth for XGBoost, num_estimators and max_depth for Random Forest. Every model will have a custom search space targeting the hyper-parameters that are usually changing to determine its functionality.

This means ensuring that the parameter ranges specified for tuning are feasible to prevent errors. In this study, hyperparameters were manually selected to avoid redundancy,

and appropriate upper bounds were applied. For example, the learning rate was set at 0.01, with a range of [0.2], and the maximum tree depth was set between 3 and 7 for XGBoost. For the RandomForest model, the number of estimators was set between 10 and 50, and the maximum depth ranged from 5 to 15. Choosing values outside these ranges could lead to inefficient searches or poor model performance. RandomizedSearchCV was also utilized with n_iter set to 5, balancing the trade-off between search time and the likelihood of finding optimal parameters.

After defining the hyperparameters, the paper train and tune the models on these (known as training data collection) using RandomizedSearchCV. For instance, different configurations get tested to find out the most appropriate model. This step will return the best hyper-parameters with the RMSE for each model.

## 4 Results Analysis

The results analysis focuses on evaluating the prediction performance and understanding the key challenges faced by the model. Three main aspects are covered: residual distribution, learning curve analysis, and the comparison between predicted and actual prices.

### 4.1 Residual Distribution

Fig. 1 shows the residual distribution of the model's predictions. The residual distribution reveals that most of the residuals are concentrated around zero, which indicates that the price predictions are more or less accurate and usually deviate not so much—for example, within [–$500, $1,000]. There is a quite sharp peak in the middle of the distribution, with over 100,000 observations near zero residual, where the model makes reasonably good predictions.

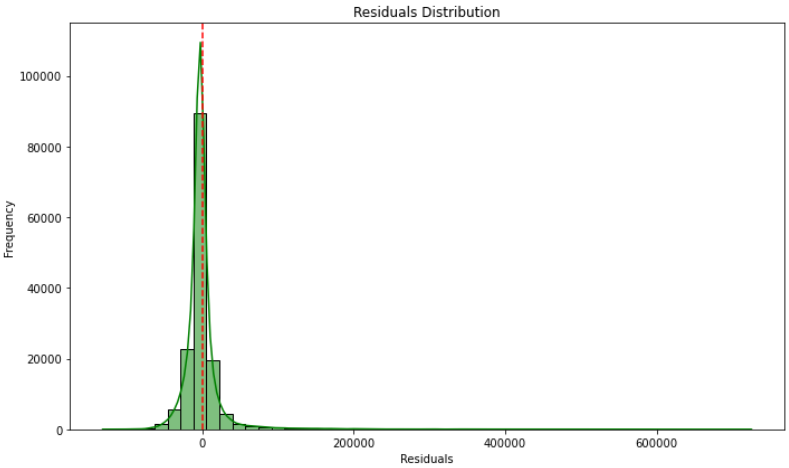


**Fig. 1.** The residual distribution of the model's predictions (Photo/Picture credit: Original )

However, there is a right-skewed tail with residuals above $200,000 and even going up to $400,000. This shows that the predictions of the model have heavily underestimated some specific samples. The outliers were possibly expensive vehicles (or rare configurations) that had different depreciation rates than the mainstream models, or there may have been poor data cleaning.

### 4.2 Learning Curve Analysis

Fig. 2 illustrates the learning curve, depicting training and validation errors as the training set size increases. The learning curve analysis indicates a training error that slowly increases with increasing size of the training set eventually leveling off at an RMSE around ca. 30,000 This indicates that the model performs well as more training instances are utilized, implying fewer overfits. The validation error too stays high, between 34,000 and 38,000 RSME throughout indicating underfitting.

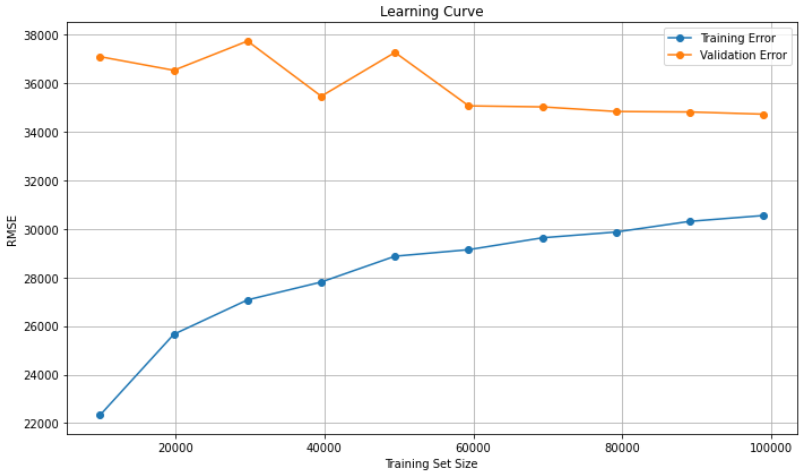


**Fig. 2.** The earning curve of the model's predictions (Photo/Picture credit: Original )

A constant gap between training and validation errors (5000–8000 RMSE) means the model fails to represent well enough all the complexity of data. That could be an indication of a too-simple model or the features were inadequate.

### 4.3 Predicted vs Actual Prices

Fig. 3 compares the predicted prices with the actual prices of the vehicles. This shows the predicted prices vs actual prices for a low-mid range of vehicle price, which means that most clusters are around the red line (which is basically when observed data points lie very close to x=y) Most of the data points are near 0-200,000 price range showing a good amount of accuracy in predicting these prices.

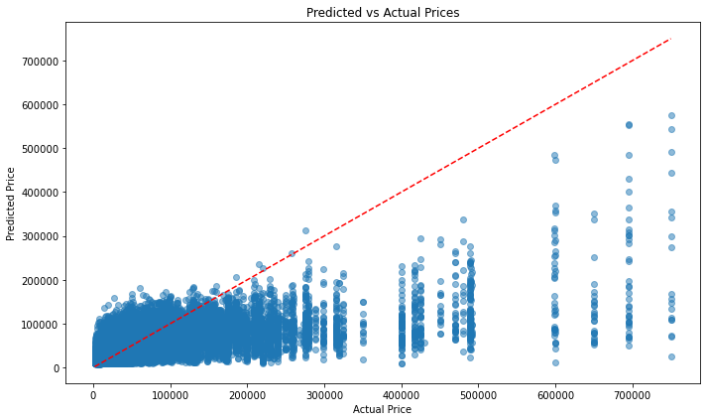


**Fig. 3.** The predicted vs actual prices of the model's predictions (Photo/Picture credit: Original )

But in the higher price range (300,000 and up), data points get more scattered but tend to be below the ideal line which means this model predicts high-priced vehicles lower prices. A reason for this could be an underestimation since the training data does not contain enough representative high-value samples or a powerful enough feature that can accurately capture some of the unique properties that drive luxury vehicle prices. Besides, The outliers on the right side of the graph may be due to the limitations in model complexity.

### 4.4 Recommendations for Improvement

Dealing with outliers is an essential first step — the residual distribution shows an extremely long tail, which indicates there are some large outliers. Hence, by improving output with an anomaly detection model or a more resilient error metric, the result could discount the appearance of these outlier values. Additionally, the learning curve suggests that there might be underfitting — and by using a more sophisticated approach or model (e.g. Gradient Boosting Machines) it should be possible to greatly decrease cross-validation error and thus improve model performance significantly in practice! Also, targeted feature engineering is an important contributor to the struggle to cover the models' under-estimating bias of high-value cars. In this case, incorporating new features reflecting luxury-specific attributes or conditional dense layer models would substantially improve the prediction performances to capture better predictions on the top segment. Finally; manually tuning feature engineering for a case with high residuals, especially high-cost vehicles can provide even better accuracy. For example, the inclusion of features around luxury specifications, brand prestige, or rarity as well as considering stratified modeling for top decile segments holds promise to yield considerable prediction accuracy gains

## 5 Conclusion

This paper explores the use of machine learning algorithms to predict second-hand car prices focusing on luxury cars. In order to achieve this, the evaluation is performed on a large dataset taken from Kaggle for which XGBoost, Random Forest, and Linear Regression methods are used. Whereas the RandomizedSearchCV method was used for tuning each of these models and based on evaluation metrics like RMSE, MAE as well as R-square the paper checked performance. Other methods like feature engineering and stratified models might also be applied to improve performance, recognizing unique attributes of luxury vehicles such as brand reputation and exclusivity that contribute meaningful information for accurate predictions.

Although conventional values are still relatively difficult to predict, the results show that even with more recent developments in traditional modeling methods for high-value vehicles; they almost always underestimated market price. Although more sophisticated models such as Random Forest and XGBoost were able to achieve enhanced performance over simpler ones by both reducing variance, which may capture the data distribution within intricate relationships of vehicle attributes with price; however, residual analysis discovered large outliers in higher-priced segments indicating that additional work on smart cropping could be done.

Moving forward, the study of Gradient Boosting as well as adding luxury-specific attributes such as brand prestige and rarity should have great potential for improvement in future research. Stratified modeling for price tiers is also interesting and may help in adding more to the predictions, especially for some high-priced vehicles[9]. Notably, the need for advanced and refined prediction models is not limited to the used car market; other

sectors, such as the P2P used car market, also highlight this demand for model development and precision[10]. In conclusion, the current study provides an important reference for researchers in the academic community and industry practitioners to optimize second-hand car price forecasts.

## Reference

1. A. Bento, K. Roth, Y. Zuo, Vehicle lifetime and scrappage behavior: Trends in the U.S. used car market. Energy J. 39(1), 159-184 (2018)
2. S. Yılmaz, İ.H. Selvi, Price prediction using web scraping and machine learning algorithms in the used car market. SAUCIS 6(2), 140–148 (2023)
3. Y. Gu, et al., An optimal sample data usage strategy to minimize overfitting and underfitting effects in regression tree models based on remotely-sensed data. Remote Sens. 8(11), 943 (2016)
4. J. Huang, et al., Integrative analysis for high-dimensional stratified models. Stat. Sin. 33, 1533-1553 (2023)
5. T. Qu, J.H. Zhang, F.T. Chan, R.S. Srivastava, M.K. Tiwari, W.Y. Park, Demand prediction and price optimization for the semi-luxury supermarket segment. Comput. Ind. Eng. 113, 91-102 (2017)
6. J. Li, H. Liu, Challenges of feature selection for big data analytics. IEEE Intell. Syst. 32(2), 9-15 (2017)
7. S. Pudaruth, Predicting the price of used cars using machine learning techniques. Int. J. Inf. Comput. Technol. 4(7), 753-764 (2014)
8. I. Guyon, A. Elisseeff, An introduction to feature extraction. In Feature Extraction: Foundations and Applications, Springer, Berlin, Heidelberg, pp. 1-25 (2006)
9. A.C. Goodman, T.G. Thibodeau, Housing market segmentation and hedonic prediction accuracy. J. Hous. Econ. 12(3), 181-201 (2003)
10. X. Ma, J. Sha, D. Wang, Y. Yu, Q. Yang, X. Niu, Study on a prediction of P2P network loan default based on the machine learning LightGBM and XGBoost algorithms according to different high dimensional data cleaning. Electron. Commer. Res. Appl. 31, 24-39 (2018)