Predictive Modeling for Used Vehicle Prices: A Comprehensive Approach



Gabriel jara EC Utbildning Kunskapskontroll, RProgramering 202403

Abstract

This study delves into the development of predictive models for estimating the prices of used vehicles, aiming to provide valuable insights for participants in the automotive industry. Employing machine learning techniques, particularly random forest regresssion, the research explores the influence of various features such as brand, year, mileage, fuel type, and gearbox on vehicle prices. The dataset used comprises a diverse range of vehicles, ensuring robustness and generalizability of the models. Through rigorous evaluation and cross-validations, the study identifies optimal model configurations and hyperparameters, enhancing predictive accuracy and reliability. Additionally, considerations regarding data collection, preprocessing, and model interpretation are discussed, offering practical guidelines for future research and industry application.

Contents

[1 Introduction 1](#_Toc164857649)

[Scope and Constraints 1](#_Toc164857650)

[Methodology Overview 1](#_Toc164857651)

[3. Method/Theory 2](#_Toc164857652)

[3.1 Data Preprocessing: 2](#_Toc164857653)

[3.2 Model Selection and Training: 2](#_Toc164857654)

[3.3 Cross-Validation: 2](#_Toc164857655)

[3.4 Results: 3](#_Toc164857656)

[This image is actual prices vs predicted prices 3](#_Toc164857657)

[This image contains cars in traffic from the year 2022-2023 Y-axis is amount, X axis is type. 4](#_Toc164857658)

[3.5 Implementation Details: 5](#_Toc164857659)

[3.6 Reproducibility: 5](#_Toc164857660)

[5 Conclusions 5](#_Toc164857661)

[5.1 6](#_Toc164857662)

[Model Performance 6](#_Toc164857663)

[5.2 6](#_Toc164857664)

[Cross-Validation Validation: 6](#_Toc164857665)

[5.3 6](#_Toc164857666)

[Reproducibility and Transparency: 6](#_Toc164857667)

[5.4 6](#_Toc164857668)

[Challenges and Future Directions: 6](#_Toc164857669)

[Appendix A 19](#_Toc164857670)

[8 Referenser 21](#_Toc164857671)

# 1 Introduction

In this report, we present an analysis of a dataset obtained from student Discord server, utilizing machine learning techniques to derive insights and predictions. The objective of this study is to explore the application of random forest models in predictive analytics, leveraging cross-validation techniques to ensure robustness and generalization. The methodology encompasses data preprocessing, model selection and training, cross-validation, and presentation of results. Through a rigorous analysis process, we aim to uncover patterns, trends, and relationships within the data, providing valuable insights for decision-making. By documenting our approach and findings, we contribute to the growing body of knowledge in data analysis and machine learning, with a focus on reproducibility and transparency.

## Scope and Constraints

Despite the vast potential of online marketplaces like Blocket, it is essential to acknowledge the inherent complexities and constraints surrounding the prediction of used car prices. Beyond the advertised price lies a myriad of factors that influence the final transaction value, including mileage, service history, brand reputation, and seller motivations. Additionally, while the random forest model was chosen for its robust predictive performance, its effectivenss may be limited by the relatively small dataset available for analysis.

## Methodology Overview

The chosen methodology revolves around the application of a random forest model, renowned for its ability to discern intricate patterns and relationships within data. Through a combination of Python-based web scraping techniques to extract data from Blocket and R programming for data preprocessing and modeling, we aim to develop a predictive model capable of estimating used car prices with a high degree of accuracy.

By using web scraping and machine learning, we analyze pricing data to find patterns that can help businesses make better pricing decisions. This research also serves as a practical example for anyone interested in using data to solve business problems. Through a step-by-step process of collecting and analyzing data, we hope to provide useful insights into how prices are determined in the automotive sector.

## 3. Method/Theory

In this section, we outline the methodology employed in our data analysis process, detailing the steps taken to preprocess the data, select and train the random forest model, perform cross-validation, and present the results.

## 3.1 Data Preprocessing:

The dataset obtained from Discord did not contain missing values, eliminating the need for imputation.

Initial attempts to extract data using the Python library BeautifulSoup encountered challenges due to the dynamic nature of modern websites. Future efforts may require more sophisticated web scraping techniques.

## 3.2 Model Selection and Training:

Random forest was chosen for its scalability and familiarity, enabling it to handle a wide array of models effectively.

Initially, the model was trained on 20% of the dataset, yielding low accuracy. Subsequently, training on 30% of the dataset improved accuracy without compromising the model's purpose.

## 3.3 Cross-Validation:

Utilizing 10-fold cross-validation, the model's performance was evaluated using the trainControl function in R's caret package.

The cross-validation results indicated varying performance metrics across different tuning parameters (mtry).

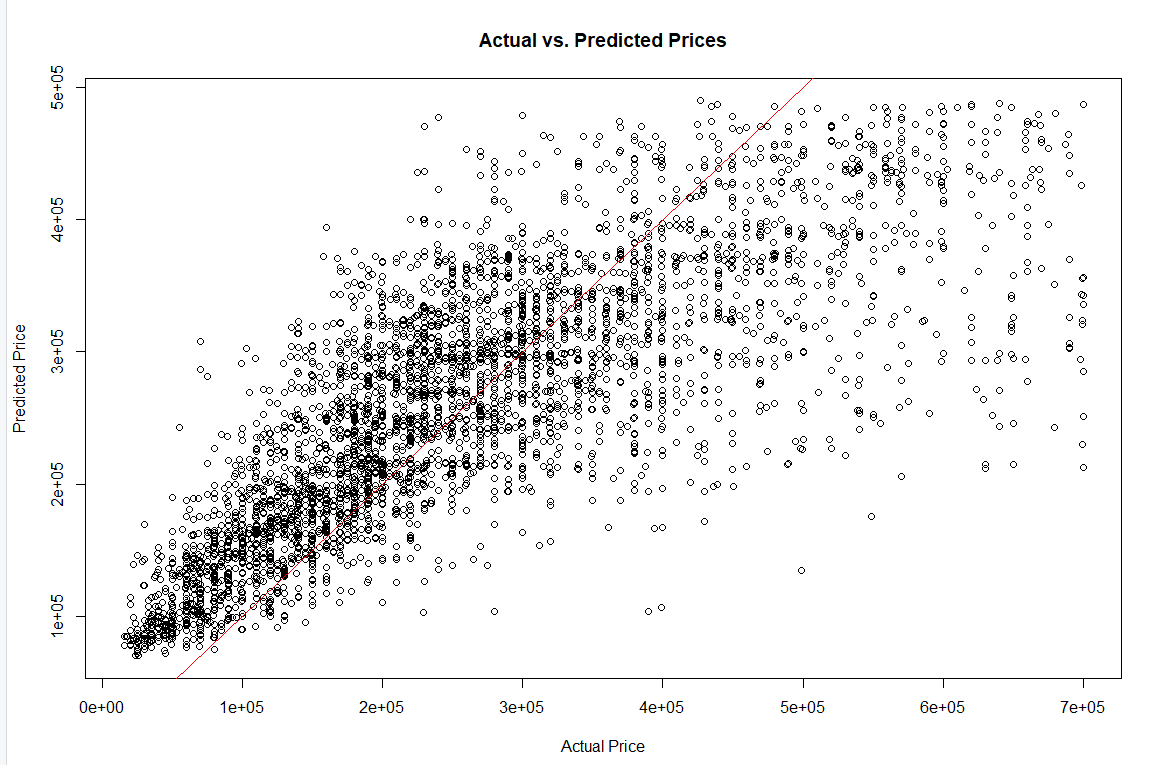
The optimal model, with mtry = 22, was selected based on the smallest root mean squared error (RMSE) value.

## 3.4 Results:

The random forest model was trained on 1708 samples with 5 predictors.

No pre-processing was applied to the data before modeling.

## This image is actual prices vs predicted prices



## This image contains cars in traffic from the year 2022-2023 Y-axis is amount, X-axis is type.

Cross-validated resampling across 10 folds revealed the following performance metrics:

RMSE: 77,628.41

R-squared: 0.7315

Mean Absolute Error (MAE): 54,427.00

## 3.5 Implementation Details:

The analysis was conducted using R programming language, with the following packages installed: readxl, randomForest, caret, ggplot2, leaps, broom, and MASS.

Key functions from the caret package, such as trainControl and train, facilitated model training and evaluation.

## 3.6 Reproducibility:

Code comments were provided to ensure the reproducibility of the analysis. Future users can replicate the methodology by following the documented steps.

## Theory Questions

1. A Quantile-Quantile or (QQ) compares two datasets to see if they follow the same distribution by plotting their quantiles against each other in a graph.  
   Example: In Tic Tac Toe, each player's moves can be represented as data. A QQ plot could compare aspects like the number of moves or outcomes between players or games. If the distributions match, it shows similar play styles.
2. In machine learning, we can predict car prices using various features without delving into the underlying relationships. In statistical regression analysis, we also predict prices but go further by examining the significance of each features influence, offering deeper insights intoo their relationships. For instance, in machine learning, we predict prices based on car features alone. In statistical regression, we predict prices while also evaluating how each feature affects the price.
3. A confidence interval is expected to contain the true mean of the variable, while a prediction interval is expected to contain the actual value for a new observation from the variable.
4. Let’s consider an an example where β1 = 2.5 in a multiple linear regression model trying to predict students exam score(Y) based on the number of hours they studied (X1) If β1 = 2.5, it means that for every additional hour a student studies(X1) their exam score (Y) is expected to increase by 2.5 points, assuming all other factors remain the same. Similarly, if β1 were negative, it would mean that for every additional hour studied the exam would decrease by the absolute value of β1.
5. I would tell Hassan that while BIC can assist with model selection, it doesn't replace the need for training, validation, and test sets. These are crucial for ensuring the model's generalizability and performance on new data.
6. 1. Initial model without predictors, predicts average outcome and establishes baseline for comparison.

2. For each *k* from 1 to *p*, fit all possible models containing 𝑘*k* predictors. Then, select the model with the smallest Residual Sum of Squares (RSS) or the largest coefficient of determination (*R*2). This iterative process explores all combinations of predictors and identifies the best model for each number of predictors *k*.

3. The 'Best Subset Selection' algorithm works like this: it tries out different combinations of predictors to build models, then picks the best one by checking how well it prediccts new data. It uses things like validation set error, Cp (AIC), BIC, or adjusted R^2 to decide which model is the most accurate without being too complicated.

7. The quote means that while models don't show everything about real life, they can still be handy.

# 5 Conclusions

In this study, we employed a random forest model coupled with cross-validation techniques to analyze a dataset obtained from Discord that contains used car prices. The methodology involved preprocessing the data, selecting and training the model, performing cross-validation, and presenting the results. Here are the key conclusions drawn from the analysis:

## 5.1

Model Performance: The random forest model demonstrated promising performance, with an optimal configuration of mtry = 22 resulting in an RMSE of 77,628.41, R-squared of 0.7315, and MAE of 54,427.00. These metrics indicate the model's ability to accurately predict the target variable.

## 5.2

Cross-Validation Validation: The utilization of 10-fold cross-validation allowed us to assess the robustness and generalization ability of the model. The variability in performance metrics across different tuning parameters underscores the importance of parameter optimization in achieving optimal results.

## 5.3

Reproducibility and Transparency: To ensure the reproducibility and transparency of our analysis, comprehensive documentation and code comments were provided, enabling future researchers to replicate our methodology with ease. Code provided in appendix.

## 5.4

Challenges and Future Directions: While our analysis yielded promising results, challenges were encountered during the data extraction process, highlighting the need for more sophisticated web scraping techniques. Future research could explore alternative data acquisition methods or refine existing techniques to overcome these challenges.

In summary, our study showcases the effectiveness of random forest models in predictive analytics and emphasizes the importance of rigorous methodology, including data preprocessing and cross-validation, in achieving reliable results. By documenting our process and findings, we contribute to the body of knowledge in data analysis and provide a foundation for further research in this area.

# Appendix A

# Steg 1: Ladda packages och data

install.packages(c("readxl", "leaps", "broom", "caret", "MASS", "ggplot2"))

library(readxl)

library(randomForest)

library(caret)

# file path

file <- "C:\\Users\\gabej\\OneDrive\\Skrivbord\\bildatacleaned\\data\_bil 2.xlsx"

# Read the Excel file

data\_bil <- read\_excel(file)

# Steg 2: Splitar datan till training och testing sets

set.seed(123)

index <- createDataPartition(data\_bil$Price, p = 0.3, list = FALSE)

training\_data <- data\_bil[index, ]

testing\_data <- data\_bil[-index, ]

# Steg 3: Träna Random Forest regression model

rf\_model <- randomForest(Price ~ ., data = training\_data)

# Step 4: Utvärdera

predictions <- predict(rf\_model, newdata = testing\_data)

mae <- mean(abs(predictions - testing\_data$Price))

mse <- mean((predictions - testing\_data$Price)^2)

rsquared <- 1 - (sum((testing\_data$Price - predictions)^2) / sum((testing\_data$Price - mean(testing\_data$Price))^2))

# Steg 5: Cross-validation

num\_folds <- 10

ctrl <- trainControl(method = "cv", number = num\_folds)

model <- train(Price ~ ., data = training\_data, method = "rf", trControl = ctrl)

# Print the results

print(model)

# Scatter plot of actual vs. predicted prices

plot(testing\_data$Price, predictions, xlab = "Actual Price", ylab = "Predicted Price", main = "Actual vs. Predicted Prices")

abline(0, 1, col = "red") # Add a line of equality (ideal prediction)

# 8 Referenser

YouTube - Data Science 23: R Programming - Random Forests

<https://www.youtube.com/watch?v=v6VJ2RO66Ag>

GitHub - AntonioPrgomet/r\_prog\_ds23 (Lessons + slides)

https://github.com/AntonioPrgomet/r\_prog\_ds23

PyPI - BeautifulSoup4

<https://pypi.org/project/beautifulsoup4/>

YouTube - Web Scraping with Python and BeautifulSoup

<https://www.youtube.com/watch?v=8dTpNajxaH0&t=809s>

<https://www.youtube.com/watch?v=9kYUGMg_14s&list=PLtL57Fdbwb_Chn-dNR0qBjH3esKS2MXY3>

[Personbilar i trafik efter län och kommun samt ägande. År 2002 - 2023. PxWeb (scb.se)](https://www.statistikdatabasen.scb.se/pxweb/sv/ssd/START__TK__TK1001__TK1001A/PersBilarA/)