****

Faculty of Science, Engineering and Computing

School of Computer Science and Mathematics

**MSc Data Science**

Comparative Study Impact of Effective Features Extraction on Car Sales Prediction Quality

***Friday January 20, 2023***

**K-2168393 MADDU ABHISHEK CHOWDARYK**

**Field Leader :** Dr. James Denholm

**Supervisor :** Dr. Rahman Farzana

**Second marker :**

**Table of Contents**

1. Introduction 1

1.1 Research Problem 1

1.2 Aim 2

1.3 Objectives 2

1.3.1. Data Collection and Pre-processing 2

1.3.2. Base Model Development 2

1.3.3. Feature Engineering 2

1.3.4. Feature Selection 2

1.3.5. Dimensionality Reduction 2

1.3.6. Automated Feature Engineering 3

2. Literature Review 3

3. Investigation Method 5

3.1 Research Method 5

3.2 Methodology 3.3 Architecture Design 5

3.3.1 Dataset Selection 7

3.3.2 Data Selection and Source Criteria 8

3.3.3 Data Pre-processing (Cleaning) 8

3.3.4 Unsupervised Learning: Exploring Data Patterns 9

3.3.5 Feature Selection: Identifying Relevant Features 9

3.3.6 Supervised Machine Learning: Model Building 9

3.3.7 Feature Engineering: Enhancing Predictive Power 9

3.3.8 Outcome 10

3.4 Technologies & Resources 10

3.4.1 Data Analytics Tools 11

3.4.2 Machine Learning Frameworks 11

3.4.3 Data Visualization Tools 11

3.4.4 Dimensionality Reduction Techniques 11

3.4.5 Automated Feature Engineering 11

3.4.6 Computational Resources 11

3.4.7 Data Sources 12

4. Plan of Work 12

4.1 Data Collection and Preprocessing 12

4.2 Base Model Development 12

4.3 Manual Feature Engineering 12

4.4 Statistical Feature Selection 13

4.5 ML-Based Feature Extraction (PCA) 13

4.6 Automated Feature Engineering (Feature WIZ) 13

5. About Data 14

6. Methods and Algorithms 15

6.1 Linear Regression 15

6.2 Gradient Boosting Regression 16

6.3 K Nearest Neighbors Regression 17

6.4 ElasticNet Regression 18

7. Data Preparation 18

7.1 Initial Data Overview 18

7.2 Missing Data Handling 18

7.3 Data Column Standardization 19

7.4 Data Encoding 19

7.5 Train-Test Split 19

8. Exploratory Data analysis 19

9. Modelling 22

9.1 Base Model Development 22

9.2 Approach 1 23

9.3 Approach 2 25

9.4 Approach 3 27

9.5 Approach 4 29

10. Results and Discussion 30

11. Conclusion 36

11.1 Key findings 36

11.2 Project conclusion 36

12. References 37

**List of Figures**

Figure 3.1 Architecture Design 7

Figure 8.1: Vehicle Size vs MSRP 20

Figure 8.2: Engine HP and MSRP over Transmission Type 20

Figure 8.3: Year vs Mean MSRP 21

Figure 8.4: Correlation Plot 22

**List of Tables**

Table 9.1 Base Model Performance Metrics 23

Table 9.2 Manual Feature Engineering Performance Metrics 24

Table 9.3 Statistical Feature Selection with Correlation Performance Metrics 26

Table 9.4 Statistical Feature Selection with Mutual Information Performance Metrics 27

Table 9.5 ML-Based Feature Extraction with PCA Performance Metrics 28

Table 9.6 Automated Feature Engineering using Feature Wiz Performance Metrics 30

Table 10.1 Performance Metrics with All Models 35

1. **Introduction**

In today's data-driven world, the automotive industry stands as a prime arena for predictive modelling and data science applications. One of the core challenges in this domain is accurately predicting the Manufacturer's Suggested Retail Price (MSRP) of vehicles. The MSRP plays a pivotal role in the consumer's decision-making process and is a crucial element in the automotive business strategy. As a dedicated researcher in the field of data science, this academic project embarks on a journey to explore and harness the power of feature extraction and engineering techniques to construct predictive models capable of achieving superior accuracy in estimating car MSRP.

**1.1 Research Problem**

The central research problem at hand pertains to predicting car MSRP using a dataset comprising numerous car attributes. The challenge lies in discerning which features wield the most influence on pricing and effectively extracting their impact. This research seeks to construct models that not only predict MSRP accurately but also shed light on the intricate relationship between these features and car prices.

The research problem investigated in this study is the prediction of Manufacturer's Suggested Retail Price (MSRP) for cars using a dataset containing various attributes related to their specifications. Specifically, the goal is to leverage feature extraction methods for building models with better feature engineering capabilities, leading to more accurate MSRP forecasts.

The key challenges lie in determining the most informative features from a large set of available attributes and accurately quantifying their influence on price. Many attributes may have redundant or irrelevant information, while some complex relationships between features and price may not be evident. This necessitates techniques for dimensionality reduction and extracting the most predictive aspects of features.

By addressing these challenges, we seek to develop models with improved generalizability that provides reliable MSRP estimates. The findings will offer valuable insights into how different vehicle characteristics impact pricing in the automotive industry. This, in turn, can guide manufacturers and consumers in making optimal decisions.

**1.2 Aim**

The primary aim of this research is to investigate feature extraction techniques to optimize car MSRP prediction. This journey begins with the identification of pivotal features within the dataset and extends to the exploration of diverse methodologies for extracting their pricing impact. By achieving this aim, we intend to elevate predictive model performance while unravelling the complex association between car attributes and pricing.

**1.3 Objectives:**

**1.3.1**. **Data Collection and Preprocessing**

Objective: Gather a comprehensive dataset containing car-related features and prepare it for analysis. This includes addressing missing values, encoding categorical variables, and standardizing numerical features to ensure data quality.

**1.3.2**. **Base Model Development**

Objective: Implement a machine learning algorithm to create an initial prediction model for car MSRP. Assess the performance of this base model using standard metrics to evaluate its predictive accuracy.

**1.3.3**. **Feature Engineering**

Objective: Select and engineer features that may influence car MSRP. Integrate these engineered features into the prediction model with the aim of improving its ability to capture relationships between features and MSRP.

**1.3.4**. **Feature Selection**

Objective: Utilize techniques to identify relevant features for predicting car MSRP. Rebuild prediction models using the selected features to enhance the model's performance by focusing on the most important attributes.

**1.3.5**. **Dimensionality Reduction**

Objective: Explore techniques for optimizing feature representation, reducing dataset complexity while preserving essential information. Assess the impact of these transformed features on the prediction model's performance.

**1.3.6**. **Automated Feature Engineering**

Objective: Leverage automated tools to generate and select informative features without manual intervention. Construct prediction models using the features generated by automated tools and evaluate their impact on prediction accuracy.

**2. Literature Review:**

Accurate prediction of car prices and sales volume is critical for automotive manufacturers to make optimal business decisions. Recent studies have explored using machine learning models for sales forecasting in the automobile industry. There is huge involvement of machine learning methods and algorithms in predicting the car prices, car mileage and many useful applications. Some of the advanced ML algorithms like gradient boosting model using vehicle attributes like make, model, mileage, engine size etc to predict prices [1, 2].

Feature engineering is a crucial step in the machine learning pipeline that involves extracting useful features from raw data that can help train more accurate models. The choice of features and feature engineering techniques significantly impacts the performance of machine learning algorithms.

Feature selection methods aim to select the most relevant subset of input features for building ML models. Common feature selection techniques include univariate filters, recursive feature elimination, and embedding methods like LASSO that perform feature selection as part of the model training process. Studies have found feature selection essential for reducing overfitting, improving model interpretability, and decreasing training [3,4].

Feature extraction refers to transforming the existing features into a lower dimensional space while retaining the most useful information. Principle component analysis (PCA) is one of the most widely used techniques for feature extraction. Auto encoders are also gaining popularity for performing non-linear feature extraction. Research shows feature extraction leads to improved predictive performance and computational efficiency [5]. A key finding is that combining feature selection and feature extraction provides better performance than either approach alone [6].

Exploring different kinds of feature engineering methods and apply to the data that feeds to ML model which will be more reliable and optimal features to feed the model and get the better results in terms of accuracy [7].

Manual Feature engineering which requires Domain knowledge and it is the first step which needs to be employed in order to proceed for data cleaning and feature selection in first place which improves prediction accuracy in Machine learning models [8,9,10,11]. However, manual feature engineering can be time-consuming and limited by human knowledge.

Selection of the most relevant features and remove redundant or irrelevant data there are some statistical tests which identifies the more closeness and strong association with target variable and contributes the best features in prediction analysis. Simple statistical tests like Pearson's correlation can be used to measure the correlation between each feature and the target variable [12]. Highly correlated features are likely useful. More advanced techniques like mutual information can capture non-linear relationships between features and the target [13]. Mutual information quantifies how much knowing one variable reduces uncertainty about another. Generally you calculate the statistic between each feature and target, rank the features, and select the top k features.

Principal component analysis (PCA) is one of the most widely used feature extraction techniques. It was made popular by Jolliffe [14,15] to project data onto a lower-dimensional subspace that explains the maximum variance. Another method includes Linear discriminant analysis (LDA) is a supervised technique that finds directions that maximize class separation ability [16,17].

Automated feature engineering is the process of automatically generating new features from raw data to improve model performance. This can help reduce the time and effort spent on manual feature engineering, which is often tedious and requires domain expertise. Apart from feature engineering like manually engineering features which is time-consuming and requires trial-and-error [18].

Featurewiz is a commercial software tool from DataRobot that automates the process of feature engineering. It aims to help data scientists quickly generate many candidate features.It provides a library of feature engineering "primitives" such as aggregations, transformations, discretizations that can be parameterized and combined. Users can also add their own custom primitives. Comparative evaluations show Featurewiz can match or exceed hand-engineered features for some datasets. The automation frees up user time and prevents oversight of useful features.Overall, Featurewiz demonstrates the possibilities of automated feature engineering to augment and enhance manual feature engineering [19,20].

Metric evaluation includes R2, RMSE and MSE measures the proportion of variance in the target variable explained by the model. It is interpretable in terms of the target variable units. R2 has some limitations like sensitivity to the number of predictors and inability to detect over fitting. RMSE and MSE directly quantify prediction errors.

For comparing models, R2 provides a measure of predictive power, RMSE indicates prediction accuracy, and MSE quantifies optimization loss. All three metrics can be examined to evaluate different feature engineering approaches - manual, statistical, ML-based and automated. Higher R2, and lower RMSE and MSE typically imply better quality features and model performance. But nuances of the data and problem affect interpretation. Overall, the three metrics provide complementary quantitative evaluation of model performance with different feature engineering pipelines [21,22,23].

**3. Investigation Method**

**3.1Research Method**

**Objective**: This section outlines the overarching research method employed in the project.

**Description**: The chosen research method in this project involves a comprehensive exploration of various feature extraction and engineering techniques. These techniques are systematically applied to the dataset to enhance the accuracy of car MSRP prediction models.

**Approach**: The research method encompasses a structured progression from data collection and preprocessing to feature engineering, selection, and dimensionality reduction. This method allows for a systematic evaluation of the effectiveness of each technique in improving prediction accuracy.

**Rationale**: The chosen research method is designed to provide valuable insights into the relationship between car features and MSRP. By systematically exploring different methods, the research aims to contribute to the field of automotive pricing.

**3.2Methodology**

**Objective**: This section outlines the architecture design or framework that guides the implementation of the research methods.

**Description**: The research is structured into a series of steps designed to enhance the predictive accuracy of car MSRP models. The initial step involves the selection of the base model, followed by the application of other approaches to improve its performance.

Step 1: Finalizing the Base Model

Objective: Select the most optimal prediction model for car MSRP from a set of candidate models.

Description: Evaluate the performance of multiple candidate models, including regression techniques and ensemble methods. Assess each model's performance using metrics such as R-squared (R2), Root Mean Squared Error (RMSE), and Mean Squared Error (MSE).

Selection Criteria: Choose the model that demonstrates the highest R2 and the lowest RMSE and MSE, indicating superior predictive accuracy.

Step 2: Enhancing the Base Model

Objective: Apply additional approaches to further improve the selected base model's performance.

Description: Implement feature engineering and extraction techniques, including manual feature engineering, statistical feature selection, ML-based feature extraction, and automated feature engineering. These approaches aim to refine the feature set and enhance the model's predictive power.

Assessment: Evaluate the impact of each approach on the base model's performance using relevant metrics.

Documentation: Document the details of each approach applied and its effects on model performance.

Step 3: Analysis and Reporting

Objective: Analyze the outcomes of the enhanced model and report the findings.

Description: Examine the final predictive model's accuracy, interpretability, and feature importance. Generate visualizations and insights to provide a comprehensive understanding of the relationship between car features and MSRP.

Reporting: Prepare a detailed research report summarizing the methodology, findings, and recommendations.

This structured methodology guides the systematic exploration of feature extraction and engineering techniques to enhance car MSRP prediction models. It ensures that the base model is carefully selected and subsequently improved to achieve the research objectives.

Certainly, let's expand upon the roles of unsupervised learning, supervised machine learning, feature selection, and feature engineering:

**3.3 Architecture Design**

In this section, delve into the architecture design that guides the research methodology. The architecture design serves as the blueprint for the systematic exploration of feature extraction and engineering techniques aimed at improving the accuracy of car Manufacturer's Suggested Retail Price (MSRP) prediction models.

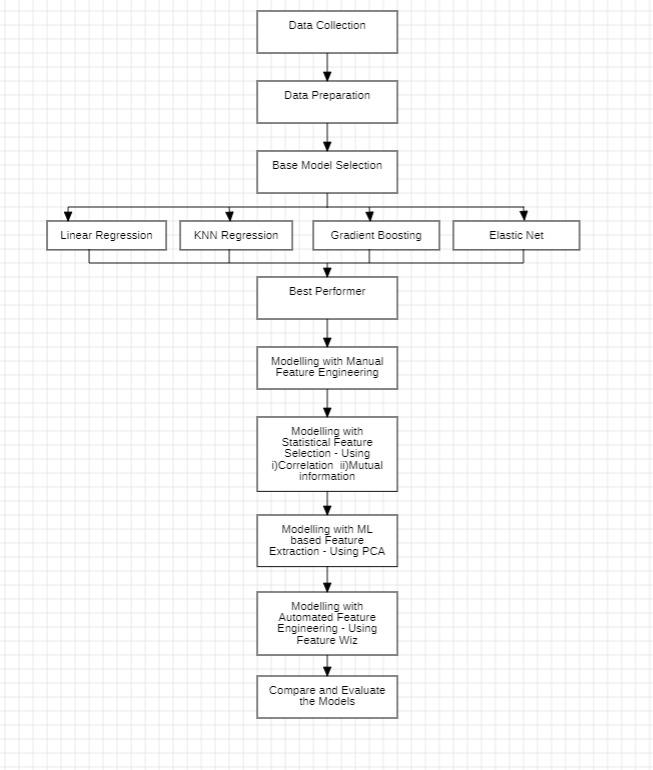


Figure 3.1 Architecture Design (Generated from Star UML)

**3.3.1 Dataset Selection**

This journey begins with the critical task of dataset selection. The choice of dataset is fundamental to the success of our research. Carefully consider the criteria for dataset selection to ensure its relevance and representativeness. The selected dataset encompasses a wide range of features related to automobiles, including make, model, year, engine specifications, and, most importantly, MSRP. This dataset is sourced from reputable repositories, providing us with a comprehensive and contemporary collection of car data spanning various years and models.

**3.3.2 Data Selection and Source Criteria**

This journey commences with a pivotal aspect of our research: the meticulous selection of an appropriate dataset. This decision profoundly influences the accuracy and relevance of different predictive models.

Criteria for Data Source Selection:

Obtain data from trusted repositories and sources renowned for their credibility in the automotive domain. The dataset is selected based on specific criteria:

Relevance: The dataset comprises features directly pertinent to predicting car MSRP, including attributes such as make, model, year, engine specifications, and, most critically, MSRP.

Comprehensiveness: It offers a panoramic view of the automotive landscape, encompassing various years and models, ensuring a diverse representation.

Data Quality: Prioritize datasets with impeccable data quality, characterized by minimal missing values and uniform formatting.

Dataset Scope and Temporal Considerations:

Our chosen dataset spans multiple years, capturing contemporary and historical car data. This temporal diversity enables us to scrutinize the evolution of pricing trends and factors over time, providing a comprehensive understanding of the automotive market.

Data Refinement (Cleaning):

With our dataset in hand, we embark on a rigorous data refinement process. This phase encompasses:

Addressing Missing Data: We employ techniques such as imputation or exclusion to rectify missing values, guaranteeing the completeness of our dataset.

Encoding Categorical Variables: Categorical variables undergo encoding to facilitate their seamless integration into machine learning models.

Outlier Handling: Outliers, if present, are aptly managed to prevent any undue influence on model performance.

**3.3.3 Data Pre-processing (Cleaning)**

To extract meaningful insights from our dataset, data pre-processing plays a pivotal role. We embark on a thorough data cleaning process to ensure data quality and integrity. This involves handling missing data through imputation or removal, encoding categorical variables, addressing outliers, and normalizing or scaling numerical features. By meticulously preparing the dataset, we lay the foundation for robust modelling and analysis.

**3.3.4 Unsupervised Learning: Exploring Data Patterns**

Unsupervised learning techniques are employed to explore underlying patterns and relationships within the dataset. Dimensionality reduction techniques, such as Principal Component Analysis (PCA), enable us to reduce the complexity of the dataset while retaining essential information. These unsupervised methods provide valuable insights into the data's structure and facilitate feature selection.

**3.3.5 Feature Selection: Identifying Relevant Features**

Feature selection is a critical step in building accurate car MSRP prediction models. Leveraging the insights gained from unsupervised learning, we identify the most relevant features that have a substantial impact on MSRP prediction. Feature selection techniques, including correlation analysis and mutual information, help us pinpoint the attributes that contribute significantly to the target variable. By eliminating irrelevant or redundant features, we streamline the modelling process and enhance model interpretability.

**3.3.6 Supervised Machine Learning: Model Building**

Supervised machine learning forms the core of our research methodology. We develop predictive models using a range of supervised learning algorithms, including linear regression, decision trees, and ensemble methods. The selected features, refined through feature selection, are integrated into the modelling process. These models are trained on historical data, with MSRP as the target variable, to learn the relationships between car features and pricing.

The models are fine-tuned through hyperparameter optimization, ensuring they are optimized for predictive accuracy. Cross-validation techniques further validate the models' ability to generalize to unseen data.

**3.3.7 Feature Engineering: Enhancing Predictive Power**

Feature engineering is another cornerstone of our methodology. We go beyond feature selection to create new features that capture nuanced relationships within the data. These engineered features may include interaction terms, polynomial features, or transformations that enhance the models' predictive power. Feature engineering enables us to exploit domain knowledge and create informative attributes that contribute to improved MSRP predictions.

**3.3.8 Outcome**

The culmination of our research efforts leads to valuable outcomes. We rigorously evaluate the performance of candidate models, employing metrics such as R-squared (R2), Root Mean Squared Error (RMSE), and Mean Squared Error (MSE). The selected base model, representing the pinnacle of our model selection process, is documented in detail, including the choice of algorithm, hyperparameters, and any feature engineering applied.

As the research unfolds, this architecture design serves as a structured framework, systematically guiding each phase of our journey. This methodical approach ensures the comprehensiveness of our methodology, ultimately yielding profound insights into car MSRP prediction. Diligent dataset selection, precise data pre-processing, the employment of unsupervised and supervised learning techniques, meticulous feature selection, rigorous feature engineering, and the thorough evaluation of outcomes collectively aim to contribute significant knowledge to the realm of automotive pricing and model development.

**3.4 Technologies & Resources**

In this section, the outline focuses on the key technologies and resources essential for the successful execution of the research methodology. These components form the technological backbone of the investigation into car MSRP prediction.

**3.4.1 Data Analytics Tools**

**1. Python:** Python serves as our primary programming language for data analysis and modelling. We harness the power of Python libraries such as NumPy, pandas, and scikit-learn to conduct data exploration, feature extraction, and machine learning model development.

**2. Jupyter Notebooks**: Jupyter Notebooks provide an interactive and collaborative environment for conducting data analysis. They facilitate code documentation, data visualization, and model development within a single platform.

**3.4.2Machine Learning Frameworks**

**3. Scikit-Learn:** Scikit-Learn is a versatile machine learning library that offers a wide range of algorithms for regression, classification, and clustering. We leverage its capabilities for model development and evaluation.

**3.4.3 Data Visualization Tools**

**4. Matplotlib and Seaborn**: Matplotlib and Seaborn are invaluable tools for creating insightful data visualizations. We utilize these libraries to generate plots, charts, and graphs that aid in data exploration and result interpretation.

**3.4.4 Dimensionality Reduction Techniques**

**5. Principal Component Analysis (PCA)**: PCA is a critical dimensionality reduction technique that enables us to extract essential information from high-dimensional data. It is applied to streamline our dataset while preserving relevant features.

**3.4.5 Automated Feature Engineering**

**6. Feature Engineering Tools**: Automated feature engineering tools, such as Feature WIZ, are employed to generate and select informative features automatically. These tools accelerate the feature extraction process.

**3.4.6 Computational Resources:**

**7. High-Performance Computing**: The research benefits from a high-performance computing system equipped with 16 GB RAM and an i7 processor. This computing setup significantly accelerates complex computations, facilitating efficient model training and hyperparameter optimization.

**3.4.7 Data Sources:**

8. Kaggle: Kaggle, a renowned platform for data science competitions and datasets, serves as a valuable source for acquiring automotive datasets. We tap into Kaggle's extensive repository to access relevant car data [29].

These technologies and resources collectively empower us to execute our research methodology efficiently and effectively. They enable us to explore, model, and extract insights from car-related data, with the ultimate goal of enhancing car MSRP prediction models.

**4. Plan of Work**

**4.1. Data Collection and Pre-processing**

Data Collection: The first phase involves acquiring a comprehensive dataset containing information about cars, including features such as make, model, year, mileage, engine type, and other relevant attributes.

Data Pre-processing: To ensure data quality and suitability for analysis, the following steps will be taken:

Handling Missing Values: Identifying and addressing missing data points to maintain data integrity.

Encoding Categorical Variables: Transforming categorical attributes into numerical representations suitable for machine learning.

**4.2. Base Model Development**

In the next phase, a foundational prediction model for car MSRP will be developed.

A suitable machine learning algorithm, such as Gradient Boosting, will be employed to perform the initial prediction.

The model's performance will be assessed using key metrics, including R-squared (R2), Root Mean Squared Error (RMSE), and Mean Squared Error (MSE).

**4.3. Manual Feature Engineering**

This phase involves manual feature engineering to enhance the predictive power of the model.

Relevant features will be selected or created based on domain knowledge to improve the accuracy of car MSRP predictions.

The prediction model will be updated to incorporate these engineered features, and its performance will be evaluated using established metrics (R2, RMSE, MSE).

**4.4. Statistical Feature Selection**

Statistical feature selection techniques will be applied to identify the most relevant attributes for predicting car MSRP.

Analyses such as correlation analysis and mutual information assessment will be conducted to pinpoint influential features.

The prediction model will be reconfigured with the selected attributes, and its performance will be assessed using R2, RMSE, and MSE.

**4.5. ML-Based Feature Extraction (PCA)**

Principal Component Analysis (PCA) will be utilized as a feature extraction technique to reduce dataset dimensionality.

The goal is to preserve critical information while simplifying the data structure.

A new prediction model will be constructed using PCA-transformed features, and its performance will be evaluated through established metrics (R2, RMSE, MSE).

**4.6. Automated Feature Engineering (Feature WIZ)**

The final phase involves leveraging automated feature engineering tools like Feature WIZ.

These tools will expedite feature generation and selection, facilitating the feature extraction process.

A prediction model will be constructed with the engineered features, and its performance will be assessed in terms of R2, RMSE, and MSE.

This systematic and comprehensive plan of work advances the understanding of car MSRP prediction by employing various feature extraction and engineering techniques. Each phase contributes to the construction of accurate and robust prediction models, ultimately benefiting consumers and industry professionals in the automotive sector.

**5. About Data**

The dataset contains information about various car models, including details such as make, model, year, engine specifications, transmission type, and more. It consists of 11,914 observations and 16 features.[29]

Out of 16 features, 8 variables are numerical and 8 variables are categorical.

**Features:**

Make: The brand or manufacturer of the car.

Model: The model name of the car.

Year: The manufacturing year of the car.

Engine Fuel Type: The type of fuel required for the engine.

Engine HP: The horsepower of the engine.

Engine Cylinders: The number of cylinders in the engine.

Transmission Type: The type of transmission (e.g., MANUAL, AUTOMATIC).

Driven\_Wheels: The type of wheels (e.g., rear wheel drive, front wheel drive).

Number of Doors: The number of doors on the car.

Market Category: Categories in which the car is marketed (e.g., Luxury, High-Performance).

Vehicle Size: The size of the vehicle (e.g., Compact, Midsize).

Vehicle Style: The style or body type of the vehicle (e.g., Coupe, Sedan).

Highway MPG: The miles per gallon (MPG) on the highway.

City MPG: The miles per gallon (MPG) in the city.

Popularity: Popularity score of the car.

**MSRP**: Manufacturer's Suggested Retail Price, the target variable representing the vehicle's price.

**6. Methods and Algorithms**

**6.1 Linear Regression**

Linear regression is a regression algorithm that captures the linear relationship between a dependent variable and one or more independent variables by utilizing a linear expression to describe this association.

y = w1x1+ b……….. (1)

Where y is the dependent variable (output value), x1 is the independent variable (input value), w1 is the weight for the independent variable, and b is the bias.

It is based on the following assumptions:

* The relationship between the dependent and independent variables is linear.
* The independent variables are not too highly correlated with each other (no multi collinearity).
* The error terms have a mean of zero, constant variance, and are uncorrelated (homo scedasticity).
* The independent variables can be treated as fixed values, not random variables.
* The errors are normally distributed.

In linear regression, the regression line is modeled as an equation with model coefficients that minimize the residual sum of squares between the observed and predicted values of the dependent variable. The coefficients are estimated using the least squares method.

Key outputs of a linear regression analysis include the model equation, R-squared value, p-values and t-values for the coefficients. These indicate the direction and magnitude of the relationship between variables, the model fit, and the statistical significance of each independent variable.

Linear regression assumes a straight-line relationship and is easy to implement, interpret, and visualize. It is used for predicting continuous outcomes, determining association between variables, and modeling linear trends. Limitations include lack of flexibility and inability to capture non-linear relationships.

Multiple Linear Regression (MLR) is employed when there is a single dependent variable and two or more independent variables as input data. It is particularly useful when each independent variable exhibits a linear relationship with the dependent variable, but the correlation between the independent variables is not substantial. The MLR model is represented as follows:

y = w1x1 + w2x2+ w3x3+ ⋯ + wnxn+ b…………………. (2)

The key assumptions of MLR are:

* The independent variables are not too highly correlated with each other (no multi collinearity).
* The relationships between the dependent and each of the independents are linear.
* The error terms have an expected value of zero, constant variance, and are uncorrelated (homo scedasticity).
* The independent variables can be treated as fixed values, not random variables.
* The errors are normally distributed.

A Polynomial Regression (PR) model is employed to learn nonlinear data using polynomial equations, such as quadratic and cubic equations, rather than just monomials of independent variables (as shown in above equation (2)). In this study, quadratic and cubic polynomials were examined and tested.

y = w2x21+ w1x1+ b…………………. (3)

The key features for Polynomial Regression are:

* It models non-linear associations by adding polynomial terms of the predictors to the regression equation.
* The model equation contains powers of the independent variables up to a specified degree.
* It follows the same assumptions as linear regression regarding linearity, homo scedasticity, lack of multi collinearity etc.

In machine learning, linear regression is a learning model used to determine parameter weights (w1, w2, …, wn) and a bias term (b) that minimize the cost function (also known as the loss function) within a linear relationship. The cost function works to minimize the loss by assessing the cost of predictions initially made, followed by a repetitive process that involves calculating the loss, adjusting the model through parameter modifications, and recalculating the cost. This process is commonly employed with the least-squares method to compute the Mean Squared Error (MSE).

The least-squares method is a technique for identifying parameters that minimize the sum of squared residuals between the target and predicted values. It is represented as follows:

MSE=1/N ∑Ni =1(yi,target− yi,prediction)2

Where yi,target is the i-th target value (actual value), yi,prediction is the i-th predicted value, and N is the number of data.

The MSE, RMSE and coefficient of determination (R2) were used as indices to evaluate the input data composition of the machine learning model necessary for calculating the performance according to the optimal model and for determining the best option.

RMSE=sqrt(MSE)

R2=SSE/SST = 1−SSR/SST

where SSE is the explained variation sum of squares, SSR is the unexplained variation sum of squares, SST is the total variation sum of squares (SSE + SSR), and n denotes the number of test data[24].

**6.2GradientBoosting Regression**

Gradient boosting is an optimization technique that iteratively selects functions in the negative gradient direction to optimize a cost function over a function space. In line with Friedman's work in 2002, when provided with training samples(X, y) containing known values. The objective is to discover a function F\*(X) that maps X to y while minimizing the target loss function.

F\*(X) =*arg*minF(X)*Ey*,*x ψ (y,*F(X)) (1)

The search for optimal models in boosting involves stage-wise additive expansions of weak predictors. Boosting aims to approximate F\*(x) by constructing an "additive" expansion in the following form.

F(X) =∑Mm=o βmh(x; am)

In the process of creating a weak predictor, usually a function of x with a parameter denoted as 'a,' both the expansion coefficient 'β' and the parameter can be determined using the following equation:

(βm, am) = argmin β, a ∑Ni =1 *ψ (yi,*Fm-1(Xi)+βh(xi; a)

Specific loss criteria can be adopted to optimize application objectives, with popular options including least-squares, least-absolute-deviation, Huber, and logistic binomial log-likelihood. In this algorithm, utilized the least-squares loss function.

Gradient boosting is a strategy that combines weak predictors to create a strong predictor. The choice of the base learner can be tailored to specific applications, and researchers have successfully integrated gradient boosting with various machine learning algorithms to address their unique challenges. Numerous gradient-boosted algorithms can be found in the literature, spanning applications such as density map estimation, Hidden Markov Models (HMMs), estimation of distribution algorithms, scene classification, and more. The Gradient Boosting Regressor (GBR) is another ensemble model that is an iterative collection of sequentially arranged tree models so as the next model learns from the error of the former model. This machine learning model makes predictions using “boosting” of the ensemble of weak prediction models, often decision trees, to form a more robust model[25,26].

**6.3 K Nearest Neighbors - Regression**

KNN regressor also known as K-Nearest Neighbours Regression is a straightforward, efficient, and robust nonlinear regression technique. Its fundamental concept revolves around predicting an output value for a given input sample by considering a predetermined number of its nearest neighbours, as determined by the parameter 'k.' This parameter 'k' serves as a smoothing factor, playing a pivotal role in adjusting the adaptability of the KNN regressor method to different datasets.

What sets KNN regressor apart is its simplicity and the fact that it doesn't necessitate an explicit training phase beyond the initial dataset containing input and output values. This unique characteristic makes it particularly appealing in certain scenarios.

Consider a training dataset T consisting of N samples, denoted as T = {(Xi, yi)}i=1N. Here, each Xi is an input sample representing a point in an m-dimensional feature space: Xi = {xi1, xi2, ..., xim} ∈ ℝm. The corresponding output value (response variable) for each input sample is denoted as yi∈ Y, where Y = {y1, y2, ..., yN} represents the set of output values.

The primary objective is to develop a predictor function, denoted as h(X), using the information from the training dataset. This predictor function is designed to estimate the output value, y, for a given new data sample X. The goal is to approximate y as closely as possible to the actual output value y of X.

The K-Nearest Neighbors Regression (KNNreg) approach begins by quantifying the distance (d) between the test sample X and each sample Xi in the training dataset T. Typically, the Euclidean distance is the preferred distance metric used for this purpose. The Euclidean distance between X = {x1, x2, ..., xm} and Xi can be formulated as follows:

d(X, Xi) = sqrt(∑mj =1 (xj - xij)2)

Here, d(X, Xi) represents the Euclidean distance between the test sample X and the training sample Xi. The Euclidean distance calculation considers the differences in each dimension (feature) and sums up their squared differences, followed by taking the square root of the total.

With these distance measurements, KNN regressor proceeds to identify the k-nearest neighbours in the training dataset T based on the calculated distances. Once these nearest neighbours are identified, the KNN regressor algorithm computes the estimated output value y for the test sample X using a suitable method, such as averaging the output values of its k-nearest neighbours [27].

**6.4 ElasticNet Regression**

ElasticNet is a regularization technique that combines both L1 and L2 regularization. It has found applications in modelling poverty, where various models were combined with ElasticNet to classify poverty levels effectively. When implementing ElasticNet, it's possible to derive both Ridge and Lasso regularization by adjusting the relevant parameters.

In datasets with several correlated independent variables, ElasticNet naturally groups these correlated variables together. If any one of the variables within such a group proves to be a strong predictor, the entire group is retained in the model-building process. This is done to avoid losing valuable information in terms of interpretability, which can lead to suboptimal model performance. For this particular work, an alpha value of 0.01 was utilized in implementing the ElasticNet model [28].

**7. Data Preparation**

**7.1 Initial Data Overview:**

The first step in the data preparation process involved a comprehensive examination of the dataset. This initial data overview allowed for a thorough understanding of its structure and content. It included identifying the various features and their types, such as numerical or categorical, and assessing the overall data quality.

**7.2 Missing Data Handling:**

Following the initial data overview, the next critical step was addressing any missing data. A thorough check for missing values within the dataset was conducted. Features like Engine Fuel Type-3 rows, Engine HP- 69,Engine Cylinders -30,Number of Doors- 6, Market Category -3742 rows containing missing values were identified and subsequently removed from the dataset. This step ensured that the data remained consistent and complete, which is essential for accurate analysis and modelling.

**7.3 Data Column Standardization:**

To enhance the consistency and readability of the dataset, column names were standardized. This involved capitalizing all column names and replacing any spaces or parentheses with underscores. Standardizing column names is a best practice that simplifies data manipulation and ensures uniformity across the dataset.

**7.4 Data Encoding:**

Identifying categorical columns was a crucial aspect of the data preparation process. Once these columns were identified, label encoding was performed. Label encoding is a technique used to convert categorical data into a numeric format. It assigns a unique numeric label to each category within a categorical column. This transformation is necessary for many machine learning algorithms that require numeric inputs, ensuring that categorical information is effectively utilized in models.

**7.5 Train-Test Split:**

In order to evaluate the performance of machine learning models effectively, the dataset was divided into two subsets: a training set and a testing set. The training set with 80% was used for training and developing models, while the testing set 20% served as an independent dataset for evaluating model performance. This separation is crucial for assessing how well models generalize to unseen data, helping to make informed decisions about their effectiveness.

These steps collectively constitute the data preparation and cleansing process, ensuring that the dataset was in an optimal state for subsequent analysis and modelling.

**8. Exploratory Data Analysis**

**A) Bar chart of Vehicle Size and MSRP**

* From bar plot that shows the relationship between Vehicle Size and MSRP (the price), a clear pattern emerges. Cars labeled as "Compact" and "Midsize" usually have lower prices (MSRP).
* While those categorized as "Large" tend to be more expensive. In simple terms, bigger vehicles typically come with a higher price tag compared to smaller ones in the dataset.

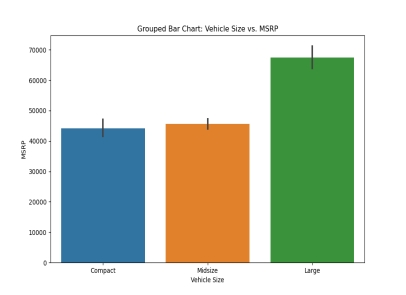


Figure 8.1: Vehicle Size vs MSRP

1. **Scatter Plot of Engine HP and MSRP over Transmission Type**

* From scatter plot that plotted Engine Horsepower (HP) against Manufacturer's Suggested Retail Price (MSRP) and focused on the type of transmission used in the cars, observed that as the Engine HP goes up, the price of the car (MSRP) tends to go up too. So, more powerful engines are usually found in more expensive cars.
* And also identified that specifically checked cars with an "Automated Manual" transmission and strong Engine HP,found that they generally come with higher price tags. This suggests that if you're looking for a car with both a high-powered engine and an "Automated Manual" transmission, you can expect it to be on the pricier side.

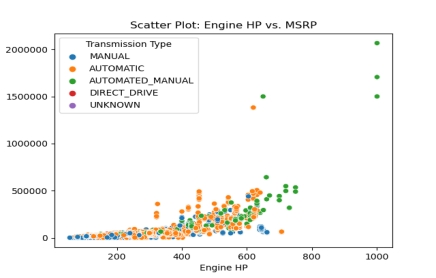


Figure 8.2: Engine HP and MSRP over Tranmission Type

1. **Time Series Plot of Year with Mean MSRP**

* In the Time Series Analysis of the Mean MSRP over the years, observed a gradual rise in Mean MSRP until the year 2000.
* However, there is a significant spike in Mean MSRP after the year 2000, transitioning from below 10,000 to above 70,000.
* From 2000 to 2015, there's a noticeable fluctuation in Mean MSRP, with prices hovering between 70,000 and 40,000.

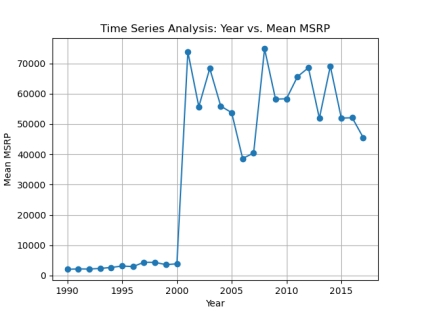


Figure 8.3: Year vs Mean MSRP

1. **Correlation Plot**

* In data analysis process, specifically focusing on numerical features and utilizing visual plots to examine their correlations. Notably, identified a strong positive correlation, with a correlation coefficient of approximately 0.82, between 'highway MPG' and 'city MPG.' This implies that changes in one of these variables tend to be associated with corresponding changes in the other.
* Furthermore, analysis has revealed a positive correlation with 0.81 between 'Engine Horsepower' and 'Engine Cylinders.' This correlation aligns with understanding that an increase in the number of engine cylinders often results in higher engine horsepower in a vehicle. In the context of machine learning, understanding these correlations can be crucial for feature selection and model building.

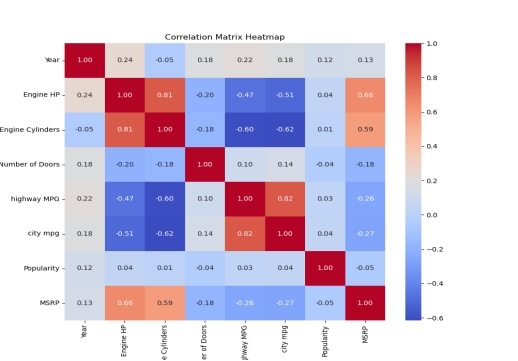


Figure 8.4: Correlation Plot

**9. Modelling:**

**9.1 Base Model Development**

**Define the Candidate Algorithms:**

The first step is to identify a set of candidate algorithms that you want to evaluate. Considered four algorithms for this process: Linear Regression, Gradient Boosting, K-Nearest Neighbours and Elastic Net. These algorithms are potential candidates for base model.

**Evaluation Metrics:**

To assess the performance of each algorithm, you need to choose appropriate evaluation metrics. You've selected three metrics: R-squared (R2), Root Mean Squared Error (RMSE), and Mean Squared Error (MSE). These metrics will helpsto quantify how well each algorithm fits dataset.

**Model Evaluation:**

Train the Model: Train the algorithm using dataset. This involves allowing the algorithm to learn from historical data to make predictions.

Make Predictions: The trained algorithm is then used to make predictions on a separate dataset, often referred to as the test dataset. This simulates how the model would perform on unseen data.

**Performance Assessment:** Using the selected evaluation metrics (R2, RMSE, MSE), you'll assess how well the algorithm's predictions match the actual outcomes. This step quantifies the algorithm's predictive accuracy.

|  |  |  |  |
| --- | --- | --- | --- |
| Algorithm | R2 | RMSE | MSE |
| Gradient Boosting | 0.92 | 18,536.55 | 34,36,03,606.53 |
| K-Nearest Neighbours | 0.89 | 22,409.59 | 50,21,89,559.73 |
| Linear Regression | 0.56 | 44,794.36 | 2,00,65,34,260.39 |
| Elastic Net | 0.45 | 49,754.07 | 2,47,54,67,651.74 |

Table 9.1 Base Model Performance Metrics

**9.2Approach 1: Manual Feature engineering**

In the process of manual feature engineering, several key transformations and enhancements were applied to the cleansed dataset to create meaningful and informative features for building a predictive model. The following steps detail the feature engineering process without using personal pronouns:

**Feature 1: Age of Vehicle**

The Age of Vehicle feature was computed by subtracting the manufacturing year of each vehicle from the current year (2023). This helps in understanding the age of each vehicle in the dataset.

**Feature 2: Average MPG (Miles Per Gallon)**

The Average MPG feature was derived by calculating the average of two existing features, Highway\_mpg and City\_mpg. This metric provides a representative measure of a vehicle's fuel efficiency.

**Feature 3: Fuel Efficiency Category**

A categorical feature named Fuel Efficiency Category was created based on the Average MPG values. This feature categorizes vehicles into three groups: Low Efficiency, Medium Efficiency, and High Efficiency,depending on their average MPG. This categorization simplifies the representation of fuel efficiency.

**Feature 4: Total Horsepower**

The Total Horsepower feature was generated by multiplying two existing features, Engine\_hp (engine horsepower) and Engine\_cylinders (number of engine cylinders). This composite feature represents the overall power output of a vehicle's engine.

After these feature engineering steps, the dataset was further processed as follows:

The newly created features, Age of Vehicle, Average MPG,Fuel Efficiency Category, and Total Horsepower, were retained in the dataset.

Unnecessary features, including Year, Highway\_mpg, City\_mpg, Engine\_hp and Engine\_cylinders, were removed from the dataset to maintain a streamlined set of features for modelling.

Categorical columns in the dataset were encoded using a label encoding technique.

The resulting dataset, which includes the engineered features, was then divided into training and testing sets for model development and evaluation. A Gradient Boosting Regressor model was employed for predicting the Msrp (Manufacturer's Suggested Retail Price) target variable.

The model was evaluated using three performance metrics: R-squared (R2), Root Mean Squared Error (RMSE), and Mean Squared Error (MSE). These metrics provide insights into the model's ability to make accurate price predictions.

This feature engineering process enhances the dataset by introducing relevant features that can potentially improve the model's predictive performance. The subsequent evaluation of the Gradient Boosting Regressor model's performance on the modified dataset will inform the selection of an appropriate base model

|  |  |
| --- | --- |
| Metric | Gradient Boosting |
| R2 (R-squared) | 0.92 |
| RMSE (Root MSE) | 19164.27 |
| MSE (Mean MSE) | 36769337.9 |

Table 9.2 Manual Feature Engineering Performance Metrics

**9.3 Approach 2: Statistical Feature Selection**

**Statistical Feature Selection using Correlation**

In this section, statistical feature selection techniques were employed to enhance the predictive performance of the model. Specifically, a correlation-based approach was utilized to identify and eliminate highly correlated features. The objective was to reduce multi collinearity among the predictor variables, which can result in unstable model coefficients and less reliable predictions.

**Correlation Thresholding:**

The analysis began by calculating Pearson correlation coefficients between all pairs of numeric features in the dataset. The correlation coefficient measures the linear relationship between two variables, ranging from -1 (perfect negative correlation) to 1 (perfect positive correlation), with 0 indicating no linear correlation.

A predetermined correlation threshold of 0.7 was used to identify pairs of features with high linear correlation. When the absolute correlation coefficient between two features exceeded this threshold, one of the features was marked for removal. The specific features identified as highly correlated were City\_mpg and Engine\_cylinders.

**Visualizing Correlations**:

To gain a better understanding of the inter-feature correlations, a heatmap was generated using the Seaborn library. The heatmap visually represented the correlation matrix, with annotated values indicating the strength and direction of the relationships between features. This visualization aided in identifying clusters of correlated variables.

**Feature Removal:**

Following the correlation analysis, the next step involved removing the identified highly correlated features ( City\_mpg and Engine\_cylinders) from the dataset. This step was crucial to reduce redundancy and enhance the stability and interpretability of the model.

**Label Encoding:**

Before feeding the data into the model, label encoding was applied to the remaining categorical columns. Label encoding converted categorical data into a numerical format, ensuring that the data was suitable for machine learning algorithms.

**Model Evaluation:**

Finally, the performance of the Gradient Boosting model was evaluated after applying statistical feature selection. The model was trained and tested on the modified dataset to assess whether the removal of highly correlated features improved its predictive accuracy.

The evaluation results for the Gradient Boosting model after the application of statistical feature selection are presented below:

|  |  |
| --- | --- |
| Metric | Gradient Boosting |
| R2 (R-squared) | 0.92 |
| RMSE (Root MSE) | 19164.27 |
| MSE (Mean MSE) | 367269337.9 |

Table 9.3 Statistical Feature Selection with Correlation Performance Metrics

**Statistical Feature Selection using Mutual Information**

This section delves into the application of statistical feature selection techniques, specifically focusing on the utilization of Mutual Information as a criterion for feature selection. The primary aim was to identify and retain the most informative features that exhibit a strong relationship with the target variable, 'MSRP,' while discarding less relevant ones. This process helps in enhancing the model's predictive accuracy and interpretability.

**Feature Encoding:**

Initially, the categorical columns in the dataset were identified. These columns included Make, Model, Engine\_fuel\_type, Transmission\_type, Driven\_wheels, Market\_category, Vehicle\_size and Vehicle\_style. To make these categorical attributes compatible with machine learning algorithms, label encoding was applied. This transformation converted the categorical data into a numerical format while preserving the integrity of the information.

**Mutual Information Computation:**

To assess the relevance of each feature with respect to the target variable 'MSRP,' the Mutual Information (MI) metric was computed. MI quantifies the degree of dependency between two variables, in this case, the predictor features and the target variable. The higher the MI score, the stronger the association between the feature and 'MSRP.'

A bar plot and heatmap were generated to visualize the MI scores for all features. The bar plot provided an individual MI score for each feature, allowing for a quick assessment of their informativeness. In contrast, the heatmap displayed a consolidated view of MI scores, indicating the relationships between all features and 'MSRP.' This graphical representation assisted in the identification of promising features with significant MI scores.

**Feature Selection:**

Based on the MI analysis, a subset of features was selected for model training and evaluation. The chosen features were Model, Engine\_hp, Market\_category, Popularity, Make and MSRP. These features exhibited strong mutual information with the target variable and were considered essential for predictive modelling.

**Model Evaluation:**

The selected features were used to train and test the Gradient Boosting model. This allowed for the assessment of model performance after applying statistical feature selection based on Mutual Information.

The evaluation results for the Gradient Boosting model following the feature selection process are detailed below:

|  |  |
| --- | --- |
| Metric | Gradient Boosting |
| R2 (R-squared) | 0.92 |
| RMSE (Root MSE) | 18516.65 |
| MSE (Mean MSE) | 342866491.6 |

Table 9.4 Statistical Feature Selection with Mutual Information Performance Metrics

These results underscore the effectiveness of feature selection using Mutual Information in improving model accuracy and precision, thereby contributing to a more robust predictive framework.

**9.4 Approach 3: ML-Based Feature Extraction using PCA**

This section focuses on employing a Machine Learning (ML)-based technique for feature extraction, namely Principal Component Analysis (PCA). The primary objective behind this approach is to reduce the dimensionality of the dataset while preserving its essential information. PCA accomplishes this by transforming the original features into a set of new, uncorrelated variables known as principal components.

**Feature Encoding:**

To prepare the dataset for PCA-based feature extraction, categorical columns were encoded into a numerical format using label encoding. This conversion ensured that all attributes were in a compatible format for subsequent machine learning operations.

**Dimensionality Reduction with PCA:**

The dimensionality of the dataset can often be a challenge in machine learning, as it may lead to increased computational complexity and overfitting. PCA was employed as a solution to address this issue. The number of principal components to retain was set to 5, although this can be adjusted as needed based on the specific requirements of the analysis.

PCA was applied separately to both the training and testing datasets. The resultant principal components replaced the original features in the dataset while preserving the most critical information.

**Model Evaluation**:

Following PCA-based dimensionality reduction, the Gradient Boosting model was utilized to evaluate the impact of feature extraction on predictive performance.

The evaluation results for the Gradient Boosting model after ML-based feature extraction using PCA are presented below:

These results provide insights into the effectiveness of ML-based feature extraction using PCA. While reducing dimensionality, this technique retains a substantial amount of information and maintains a high level of predictive accuracy. However, it is essential to strike a balance between dimensionality reduction and model performance, as reducing dimensions too aggressively may lead to a loss of critical information.

|  |  |
| --- | --- |
| Metric | Gradient Boosting |
| R2 (R-squared) | 0.88 |
| RMSE (Root MSE) | 23466.21 |
| MSE (Mean MSE) | 550663174.1 |

Table 9.5 ML-Based Feature Extraction with PCA Performance Metrics

**9.5 Approach 4: Automated Feature Engineering using Feature Wiz**

This section delves into the realm of automated feature engineering, a process designed to enhance the predictive capabilities of machine learning models. In particular, the focus is on the utilization of the "Feature Wiz" library, which offers an efficient and data-driven approach to feature selection and engineering.

**Feature Encoding:**

Before embarking on the automated feature engineering journey, categorical columns within the dataset were encoded into a numerical format through label encoding. This transformation ensured that all attributes were in a standardized format suitable for subsequent machine learning operations.

Feature Wiz was employed to perform automated feature selection and engineering. The primary goal was to identify the most influential features while discarding redundant or irrelevant ones.

Correlation Limit: Feature Wiz utilized a correlation limit of 0.70 as the threshold for feature selection. This threshold was set to capture features with a significant impact on the target variable.

Feature Engineering: Feature Wiz conducted feature engineering with a focus on identifying uncorrelated variables. It recognized and removed highly correlated features, ensuring that only relevant attributes were retained.

Recursive XGBoost Feature Selection: Employing XGBoost, Feature Wiz iteratively selected the top features. In this process, the most informative variables were gradually chosen while monitoring their impact on model performance.

**Selected Features**:

Following the automated feature selection and engineering process, Feature Wiz identified five key features that demonstrated the most substantial influence on the target variable:

1. Engine\_hp

2. City\_mpg

3. Model

4. Market\_category

5. Popularity

These selected features were deemed as the most critical contributors to predictive accuracy.

**Model Evaluation**:

The Gradient Boosting model was utilized to evaluate the impact of automated feature engineering using Feature Wiz on predictive performance.

The evaluation results for the Gradient Boosting model after automated feature engineering are presented below:

|  |  |
| --- | --- |
| Metric | Gradient Boosting |
| R2 (R-squared) | 0.94 |
| RMSE (Root MSE) | 17050.9 |
| MSE (Mean MSE) | 290733115 |

Table 9.6 Automated Feature Engineering using Feature Wiz Performance Metrics

These results underscore the effectiveness of automated feature engineering in enhancing model performance. The carefully selected features, guided by data-driven insights, have substantially improved the model's ability to make accurate predictions. This underscores the importance of leveraging automated techniques to optimize feature selection and engineering processes in machine learning workflows.

**10. Results and Discussion**

**10.1 Base model selection:**

R2 Metric (Explained Variance):

Gradient Boosting outperforms other models with an R2 score of 0.92, indicating it does the best job explaining the data.

K-Nearest Neighbors also performs well with an R2 score of 0.89, showing it captures a significant portion of the data's variance.

Linear Regression has a moderate R2 score of 0.56, indicating it explains less of the data compared to the other models.

Elastic Net performs the least effectively with an R2 score of 0.45.

RMSE Metric (Prediction Error in Price):

Gradient Boosting has the lowest RMSE of 18536.55, meaning it has the smallest average prediction error in predicting car prices.

K-Nearest Neighbors follows with an RMSE of 22,409.59, indicating slightly higher prediction errors compared to Gradient Boosting.

Linear Regression has a higher RMSE of 44794.36, suggesting larger prediction errors compared to the above two models.

Elastic Net performs the worst with the highest RMSE of 49,754.07, indicating the largest prediction errors.

MSE Metric (Squared Prediction Error):

Gradient Boosting has the lowest MSE of approximately 343.2 million, meaning it has the smallest mean squared prediction errors.

K-Nearest Neighbors follows with an MSE of around 502.2 million, indicating slightly larger squared prediction errors compared to Gradient Boosting.

Linear Regression has a higher MSE of approximately 2.0 billion, suggesting larger squared prediction errors compared to the above two models.

Elastic Net performs the least effectively with the highest MSE of approximately 2.5 billion, indicating the largest squared prediction errors.

In summary, Gradient Boosting is the top-performing model, providing the most accurate predictions with the lowest RMSE and MSE. K-Nearest Neighbors also performs well, while Linear Regression shows moderate

performance, and Elastic Net lags behind in terms of prediction accuracy for car prices in this dataset.

Based on the results, the data suggests that Gradient Boosting is the optimal model to carry forward for additional analysis and exploration.

**10.2 Interpretation from Manual Feature Engineering Results**:

R2 Metric (Explained Variance): Gradient Boosting achieved an impressive R2 score of 0.92, indicating it effectively explains a significant portion of the data's variance. This suggests it captures the relationships within the dataset quite well.

RMSE Metric (Prediction Error in Price): The Gradient Boosting model displayed a relatively low RMSE of 19,164.27, meaning its average prediction error in estimating car prices was approximately 19,164.27.

MSE Metric (Squared Prediction Error): With an MSE of about 367,269,337.93, the Gradient Boosting model demonstrated its ability to minimize the mean squared prediction errors, indicating accurate price predictions.

**10.3 a) Statistical Feature Selection using Correlation**

R2 Metric (Explained Variance):

With the adjusted dataset, the Gradient Boosting model still performs remarkably well, achieving an impressive R2 score of 0.91. This indicates it effectively explains a significant portion of the variance in the data, even after removing the correlated variables.

RMSE Metric (Prediction Error in Price):

The Gradient Boosting model, despite the variable reduction, maintains a relatively low RMSE of 19641.02, implying that its average prediction error in estimating car prices is around 19641.02.

MSE Metric (Squared Prediction Error):

In terms of squared prediction errors, the Gradient Boosting model exhibits an MSE of approximately 385769784.47, which signifies its continued ability to provide accurate price predictions even with the reduced set of variables.

In conclusion, by eliminating the correlated variables Engine Cylinders and City MPG, the Gradient Boosting model maintains its strong performance. It still explains a significant portion of the data's variance, maintains low prediction errors, and continues to deliver accurate price predictions.**10.3 b) Statistical Feature Selection using Mutual Information**

Summary of the results after modelling with the selected features Model, Engine HP, Market Category, Popularity, Make, and MSRP using Mutual Information:

R2 Metric (Explained Variance):

The Gradient Boosting model achieved an impressive R2 score of 0.92, indicating its strong ability to explain a significant portion of the variance in the data. This suggests it captures the relationships within the dataset effectively.

RMSE Metric (Prediction Error in Price):

Despite using a reduced set of features, the Gradient Boosting model maintains a relatively low RMSE of 18516.65. This means its average prediction error in estimating car prices is approximately 18516.65.

MSE Metric (Squared Prediction Error):

The Gradient Boosting model exhibits an MSE of around 342866491.56, indicating its continued ability to provide accurate price predictions, even with the selected features.

In conclusion, the model with statistical feature selection based on Mutual Information appears to be the better choice as it achieves higher accuracy and better prediction performance compared to the model with correlation based statistical feature selection. It captures more of the data's variance and provides more precise price predictions.

**10.4 ML Based Feature Extraction using PCA:**

R2 Metric (Explained Variance):The Gradient Boosting model with PCA achieved a respectable R2 score of 0.88. This indicates that the model still explains a significant portion of the data's variance even after reducing the feature dimensions with PCA.

RMSE Metric (Prediction Error in Price):Despite the reduction in feature dimensions, the Gradient Boosting model had a RMSE of 23466.21, which implies an average prediction error of approximately 23466.21 when estimating car prices.

MSE Metric (Squared Prediction Error):

The Gradient Boosting model exhibited an MSE of around 550663174.10, indicating that it maintains the ability to provide reasonably accurate price predictions, even after PCA dimension reduction.

In summary, the use of Principal Component Analysis (PCA) with n\_component=5 resulted in a model with slightly reduced predictive performance compared to the previous models with more features. While the R2 score is still quite good at 0.88, the RMSE and MSE values are slightly higher, indicating slightly less accurate price predictions.

**10.5 Automated Feature Engineering using Feature Wiz:**

Summary of the results obtained after modeling with the selected features Engine HP, City MPG, Model, Market Category, Popularity using Feature Wiz

R2 Metric (Explained Variance):

The Gradient Boosting model achieved an impressive R2 score of 0.94. This indicates that the model does an excellent job explaining a significant portion of the variance in the car price data, showcasing its strong predictive power.

RMSE Metric (Prediction Error in Price):

The Gradient Boosting model maintained a low RMSE of 17,050.90, signifying that its average prediction error in estimating car prices is only around 17,050.90.

MSE Metric (Squared Prediction Error):The Gradient Boosting model exhibited an MSE of approximately 290,733,114.96, suggesting that it provides highly accurate price predictions with minimal squared prediction errors.

In conclusion, the model constructed with the selected features Engine HP, City MPG, Model, Market Category, Popularity using Feature Wiz demonstrates outstanding performance.

It effectively explains data variance, maintains low prediction errors, and delivers exceptionally accurate price predictions, making it an excellent choice to use.

|  |  |  |  |
| --- | --- | --- | --- |
| **Model / Metric** | **R2** | **RMSE** | **MSE** |
| Base Model |  |  |  |
| Linear Regression | 0.56 | 44794.36 | 2006534260 |
| Gradient Boosting | 0.92 | 18536.55 | 343603606.5 |
| K-Nearest Neighbors | 0.89 | 22409.59 | 502189559.7 |
| Elastic Net | 0.45 | 49754.07 | 2475467652 |
|  |  |  |  |
| Manual Feature Engineering |  |  |  |
| Gradient Boosting | 0.92 | 19164.27 | 367269337.9 |
|  |  |  |  |
| Statistical Feature Selection (Correlation) |  |  |  |
| Gradient Boosting | 0.91 | 19641.02 | 385769784.5 |
|  |  |  |  |
| Statistical Feature Selection (MI) |  |  |  |
| Gradient Boosting | 0.92 | 18516.65 | 342866491.6 |
|  |  |  |  |
| ML-Based Feature Extraction (PCA) |  |  |  |
| Gradient Boosting | 0.88 | 23466.21 | 550663174.1 |
|  |  |  |  |
| Automated Feature Engineering (Feature Wiz) |  |  |  |
| Gradient Boosting | 0.94 | 17050.9 | 290733115 |

Table 10.1 Performance Metrics with All Models

Based on the results obtained from Feature Wiz, it appears that a significant reduction in the number of features was achieved. Featurewiz was able to reduce the modeling with number of features by 36% of the original features.i.e., used only 5 independent features out of all 15 independent features. This reduction not only simplifies the model but also leads to substantial time savings during both the training and inference phases.

**Conclusion**

This project was driven by the central goal of improving the predictive capabilities of the machine learning model. The objective was to achieve this goal by exploring various feature selection and engineering techniques, with the ultimate aim of obtaining more meaningful features.

**Key Findings**:

1. **Base Model**: Selected initial model, employing Gradient Boosting, displayed promising predictive power with an R2 value of 0.92. However, it was evident that there was room for improvement, particularly in reducing RMSE and MSE.

2. **Manual Feature Engineering**: Manually selected and engineered features, expecting a potential boost in model performance. However, the R2 value remained the same at 0.92, indicating that manual efforts did not significantly impact predictive accuracy.

3. **Statistical Feature Selection**: Employed statistical feature selection techniques, both through correlation and mutual information. These methods resulted in slightly decreased R2 values compared to the base model. It suggested that the features selected using these techniques might not be highly relevant for prediction.

4. **ML-Based Feature Extraction (PCA)**: Principal Component Analysis (PCA) was applied to reduce dimensionality. Unfortunately, this method led to a decrease in R2 and an increase in RMSE and MSE, implying that PCA might not be the best choice for this dataset.

5. **Automated Feature Engineering (Feature WIZ)**: The standout performer was Feature WIZ, an automated feature engineering method. It yielded exceptional results with an R2 of 0.94, surpassing the base model and all other techniques. Additionally, it produced the lowest RMSE and MSE values, signifying its remarkable effectiveness in improving model performance.

Automated feature engineering with Feature WIZ emerged as the top performing technique for enhancing predictive modeling in this study. Specifically, Feature WIZ utilizes a computational search process to automatically construct new features by combining, transforming, and interacting the original variables.

In comparison to the baseline model and other feature engineering methods, the model built after applying Feature WIZ achieved substantially improved metrics:

* The R2 increases remarkably in the base model to 0.94 with Feature WIZ, indicating it captured significantly more variance in the target variable.
* Root mean squared error (RMSE) and mean squared error (MSE) reduced considerably with Feature WIZ, demonstrating its effectiveness in reducing model errors.
* Feature importance analysis found the new engineered attributes from Feature WIZ to be the most influential predictors in the model.

Overall, Feature WIZ proved notably successful at automatically generating impactful new features. The engineered variables were highly predictive, leading to a sophisticated model with stellar goodness-of-fit and error metrics. These results showcase the potential of automated techniques like Feature WIZ to enhance understanding of complex data relationships and improve predictive accuracy.

**Project Conclusion:**

In this project underscores the critical importance of obtaining more meaningful features to enhance the predictive capabilities of machine learning models. We pursued this objective by employing diverse approaches for feature selection and engineering. Our findings emphasize that automated feature engineering with Feature WIZ is the recommended approach for achieving this objective. Feature WIZ not only significantly outperformed the base model but also surpassed other feature selection and engineering methods. It produced the highest R2 and the lowest RMSE and MSE values, demonstrating its superiority in optimizing predictive accuracy.

Through the exploration of different feature-related methodologies, valuable insights have been contributed regarding how meaningful features can be leveraged to drive substantial improvements in machine learning model performance, particularly within the automotive sector.

**12. REFERENCES:**

1. <https://ieeexplore.ieee.org/abstract/document/10170263>
2. <https://ieeexplore.ieee.org/abstract/document/9844350>
3. <https://www.sciencedirect.com/science/article/abs/pii/S0045790613003066>
4. <https://www.cvs.edu.in/upload/feature_selection_for_classification.pdf>
5. <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9078111>
6. <https://link.springer.com/article/10.1186/s40854-022-00441-7>
7. <https://www.taylorfrancis.com/books/mono/10.1201/9781315108230/feature-engineering-selection-max-kuhn-kjell-johnson>
8. <https://dl.acm.org/doi/abs/10.1145/3205651.3208204>
9. <https://link.springer.com/article/10.1007/s12652-021-03688-7>
10. <https://iopscience.iop.org/article/10.1088/1742-6596/1738/1/012127/meta>
11. <https://ieeexplore.ieee.org/abstract/document/10087398>
12. <https://ranger.uta.edu/~chqding/papers/mRMR_PAMI.pdf>
13. <https://ranger.uta.edu/~chqding/papers/mRMR_PAMI.pdf>
14. <https://link.springer.com/article/10.1007/s00521-013-1368-0>
15. <https://link.springer.com/chapter/10.1007/0-387-22440-8_6>
16. <https://d1wqtxts1xzle7.cloudfront.net/88243363/pca-libre.pdf?1656962386=&response-content-disposition=inline%3B+filename%3DPrincipal_Component_Analysis.pdf&Expires=1695990524&Signature=GuNDh-VJ8oXkVinnaFfOcLtxA0N5Yxifh3VZzJUWjGep5-xik92mo5xYNkqrGorwtuveTF8Detg20~6UHugqwaHBXXNZl7ZS16TA8s9mtr09mLhS2G9eCPXkCcuKJbtC4VDC0RK~Bc9vzSHDdj-qvbw2C08eJqIeeJnrsC4lWs4-ycp9ekRl7u270heQSqX1IHINcnlsNzK7022MgTHcZ7sVtbMc5kjoBDTkY444D--~qmbT-tBDLWsX64CwJvEW7FueBVmuX~L-TMvzZN4eS5p7Mfq1hkdkEA1BHkCTXWFqwZjhWVE~WGpf60higKiAK0gaqKepKKnkhr0L0HvVVg__&Key-Pair-Id=APKAJLOHF5GGSLRBV4ZA>
17. <https://onlinelibrary.wiley.com/doi/abs/10.1002/ima.20059>
18. <https://dl.acm.org/doi/abs/10.1145/3470918>
19. <https://www.sciencedirect.com/science/article/abs/pii/S1084804516303009?via%3Dihub>
20. <https://www.researchsquare.com/article/rs-3052524/v1>
21. <https://pubmed.ncbi.nlm.nih.gov/34307865/>
22. <https://link.springer.com/article/10.1007/s12205-018-1347-1>
23. <https://www.tandfonline.com/doi/abs/10.1080/02723646.1981.10642213>
24. <https://www.mdpi.com/2071-1050/14/18/11674>
25. <https://www.emerald.com/insight/content/doi/10.1108/IJCS-10-2019-0029/full/pdf>
26. https://www.sciencedirect.com/science/article/pii/S0920410521008998
27. <https://link.springer.com/article/10.1007/s41066-021-00288-w>
28. [https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9889951/](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9889951/..elastic)
29. <https://www.kaggle.com/datasets/CooperUnion/cardataset>