CAR PRICE PREDICTION

Submitted to Bharathiar University in partial fulfilment for the requirement of the degree of

BACHELOR OF COMPUTER SCIENCE (ARTIFICIAL INTELLIGENCE)

Submitted by,

VASANTH.V.M

(Reg.No: 2228B0077)

Under the Guidance of

Mrs.K.SANGEETHA MCA

Assistant Professor
Department of Computer Science

DEPARTMENT OF COMPUTER SCIENCE

PPG COLLEGE OF ARTS AND SCIENCE

(Affiiliated to Bharathiar University)

SARAVANAMPATTI, COIMBATORE, 641 035.

OCT-2024

CERTIFICATE

BONAFIDE CERTIFICATE

This is the certify that to the project entitled **"CAR PRICE PREDICTION"** in the bonafide work of **VASANTH.V.M(Reg No:2328F0026)** submitted to Bharathiar University in partial fulfilment of the requirement of the award for degree of Bachelor of Science in **COMPUTER SCIENCE ARTIFICIAL INTELLIGENCE** and that this work has not formed the basis of the award of any degree Associate ship or any other title of any other university.

Signature of the Guide	Signature of the HOD
Mrs.K.SANGEETHA MCA	Mrs.K.YASOTHA MCA.,M.Phil.,(Ph.D)
Assistant Professor	Head, Assistant Professor
Department of Computer Science	Department of Computer Science
PPG college of Arts and Science	PPG college of Arts and Science

Viva – Voce examination held on ________________

Internal Examiner **External Examiner**

DECLARATION

DECLARATION

I hereby declare that the project entitled **"CAR PRICE PREDICTION"** by **VASANTH.V.M** **(Reg No:2328F0026)** in partial fulfilment of the requirement of the award of degree of Bachelor Of Science in Computer Science Artificial Intelligence at PPG College of Arts and Science in an authentic record of my carried out under the supervision of **Mrs.K.SANGEETHA MCA** Assistant Professor, Department of Computer Science. The matter presented has not been submitted by me in any other university/institution for the award of degree of B.Sc. Computer Science with Data Analytics.

Place : Coimbatore

Date :

Signature of the candidate

VASANTH.V.M

(Reg No:2328F0026)

ACKNOWLEDGMENT

ACKNOWLEDGEMENT

I wish to convey my heartfelt to our honourable chairman **Dr. L. P. THANGAVELU M.S.,FAIS, FIAGES, FICS, Founder-chairman, PPG Group of Institutions**, who has always encouraged me to give for the best.

At this pleasing moment of having successfully completed out project I wish to convey my sincere thanks and gratitude to our beloved correspondent **Mrs. SHANTHI THANGAVELU** and our trustee **Mr. AKSHAY THANGAVELU** who provide all the facilities to me.

I wish to express my deep gratitude to our beloved **Dr. N. MUTHUMANI MSC.,MPhil.,Ph.D.,NET** and thank her profusely for all the motivation and mortal support.

I heartily thanks **Mrs .K. YASOTHA MCA.,M.Phil.,(Ph.D)** Head of the Department and **Mrs.K. SANGEETHA MCA** my project guide for providing me all the necessary facilities tocomplete my project successfully.

I express my thanks to other teaching and non-teaching faculty members of computer science department of **PPG College of Arts and Science** for their valuable support and co-operation while working on this project.

Finally, I am greatful to my parents and my friends for showing keen interest and for their encouragement during the course of the project work.

TABLE OF CONTENT

TABLE OF CONTENT

CHAPTER NO.	TITLE	PAGE NO.
	SYNOPSIS	
1	**INTRODUCTION**	
	1.1 ABOUT THE PROJECT	
	1.1.1 PROBLEM STATEMENT	
	1.1.2 OBJECTIVE OF THE PROJECT	
	1.2 SYSTEM SPECIFICATION	
	1.2.1 HARDWARE SPECIFICATION	
	1.2.2 SOFTWARE SPECIFICATION	
	1.2.3 SOFTWARE ENVIRONMENT	
2	**SYSTEM STUDY**	
	2.1 EXISTING SYSTEM	
	2.1.1 DRAW BACK OF EXISTING SYSTEM	
	2.2 PROPOSED SYSTEM	
	2.2.1 FEATURE OF PROPOSED SYSTEM	
3	**SYSTEM DESIGN AND DEVELOPMENT**	
	3.1 FILE DESIGN	
	3.2 DESCRIPTION OF MODULES	
	3.3 ANALYSIS	
	3.4 DATAFLOW DIAGRAM	
	3.4.1 TABLE DESIGN	
4	**DATA VISUALIZATION**	
5	**IMPLEMENTATION**	
6	**CONCLUSION**	
7	**FUTURE ENHANCEMENT**	
8	**BIBLIOGRAPHY**	
	BOOK REFERENCE	
	WEBSITE REFERENCE	
9	**APPENDICES**	
	A. SCREENSHOTS	
	B. CODING	

SYNOPSIS

Car Price Prediction" is a predictive modeling project aimed at developing a system to accurately forecast the prices of cars based on various features and factors. The project begins with data collection from various sources, including online car marketplaces, dealerships, and car manufacturers. The dataset comprises attributes such as car make, model, year, mileage, engine size, fuel type, transmission type, and several others. After collecting the data, the next step involves data preprocessing, which includes handling missing values, removing outliers, and encoding categorical variables. Following System developments a comprehensive process that encompasses the design, implementation, testing, deployment, and maintenance of software solutions to address specific needs or requirements. It involves a series of interrelated activities and stages, each aimed at achieving the overarching goal of delivering a high-quality, reliable, and scalable system that meets the needs of its users. From conceptualization to execution, system development requires careful planning, collaboration, and iteration to ensure success.

The first phase of system development is typically the requirements analysis and gathering stage. During this phase, stakeholders identify and document the functional and non-functional requirements of the system, including user needs, business objectives, and technical constraints. This involves conducting interviews, surveys, and workshops to gather input from end users, domain experts, and other relevant parties. The goal is to establish a clear understanding of the problem domain and define the scope of the system to be developed. Once the requirements are gathered, the next phase involves system design and architecture.

Here i used a Jupiter notebook to predicit a car price. The websites will be developed using Kaggle platform and the back end will managed in the python. The application will have easy and feasible GUI for all type of users

INTRODUCTION

ABOUT THE PROJECT

The "Car Price Prediction" project is a data science initiative aimed at developing accurate predictive models to estimate the prices of cars in the automotive market. With the proliferation of online car sales platforms and the increasing complexity of factors influencing car prices, there is a growing demand for reliable methods to determine fair market values. This project seeks to address this demand by leveraging advanced machine learning techniques and extensive datasets to create predictive models capable of estimating car prices with precision. The project begins with the acquisition of comprehensive datasets containing information on various attributes of cars, such as make, model, year of manufacture, mileage, engine specifications, fuel type, transmission type, and geographical location. Data is sourced from a variety of sources including online marketplaces, dealership records, and manufacturer databases. The collected data undergoes thorough preprocessing to ensure its quality and suitability for analysis. This involves tasks such as handling missing values, removing duplicates, encoding categorical variables, and scaling numerical features.

Data preprocessing is essential for preparing the dataset for model training and evaluation: Multiple machine learning algorithms are explored and evaluated for their effectiveness in predicting car prices. These may include linear regression, decision trees, random forests, support vector machines, and gradient boosting machines, among others. Each algorithm is trained using a portion of the dataset and evaluated using appropriate performance metrics to assess its predictive accuracy. The trained models are evaluated using validation techniques such as cross-validation and holdout validation to assess their generalization performance.

1.2. SYSTEM SPECIFICATION

1.2.1 HARDWARE SPECIFICIATION

HARD DISK : 512 GB SSD 1T HDD

RAM :8 GB DDR2

PROCESSOR :intel core and above ryzen 3

OPRATING SYSTEM : windows

1.2.2 SOFTWARE SPECIFICATIONS

BACK END :python

ML MODEL :numpy,pandas,SKlearn,

TOOLS :jupyter notebook

2. SYSTEM STUDY

2.1 EXISTING SYSTEM:

The existing system for car prediction involves the use of machine learning algorithms to analyze historical data related to various car attributes and customer preferences. This data typically includes features such as car make, model, year, mileage, price, fuel efficiency, and customer reviews. By applying machine learning techniques such as regression, classification, or clustering, the system learns patterns and relationships within the data to make predictions about future car sales, trends, or customer preferences. These predictions can be used by car manufacturers, dealerships, or online platforms to optimize inventory management, marketing strategies, and customer satisfaction. The existing system may also incorporate data from external sources such as market trends, economic indicators, and demographic information to improve the accuracy of its predictions. Overall, the goal of the existing system is to leverage data-driven insights to enhance decision-making processes in the automotive industry and ultimately improve business outcomes.

DRAWBACK FOR EXISTING SYSTEM

One drawback for car price prediction models could be the volatility of the used car market, which can be influenced by various factors such as economic conditions, consumer preferences, and technological advancements. This volatility can make it challenging for models to accurately predict prices over..

Lack of scalability: Existing systems may struggle to handle increased workload or user demands, leading to performance issues.

Limited flexibility: Many systems are rigid and difficult to adapt to changing requirements or technological advancements.

Dependency on legacy technology: Older systems may rely on outdated technology, making maintenance and updates challenging.

Security vulnerabilities: Vulnerabilities in existing systems can expose sensitive data to cyber threats and breaches.

2.2 PROPOSED SYSTEM:

The proposed system for car prediction is a cutting-edge solution designed to revolutionize decision-making processes within the automotive industry. By leveraging state-of-the-art machine learning algorithms and advanced data analytics techniques, this system aims to provide unparalleled insights into market trends, consumer preferences, and future demand for automobiles. At its core, the proposed system operates on the principle of predictive analytics, utilizing historical data and real-time information to forecast sales, identify emerging trends, and optimize inventory management strategies. Central to the functionality of the proposed system is its ability to ingest and analyze vast amounts of data from diverse sources. These data sources encompass a wide range of variables including car attributes (such as make, model, year, mileage, price, and features), customer demographics, market trends, economic indicators, and regulatory factors. By aggregating and harmonizing data from disparate sources, the system creates a comprehensive dataset that serves as the foundation for predictive modeling and analysis. Machine learning algorithms play a pivotal role in the predictive capabilities of the proposed system.

2.2.1 FEATURES FOR PROPOSED SYSTEM:

The proposed system will incorporate a range of features to ensure its effectiveness and user satisfaction. It will prioritize scalability, allowing seamless expansion to accommodate growing data volumes and user bases. Flexibility will be a key aspect, enabling users to customize and configure the system according to their specific requirements. Modular architecture will facilitate easy integration of new functionalities and components, ensuring adaptability to evolving needs. Security measures will be robust, safeguarding sensitive data against unauthorized access or breaches. A user-friendly interface will enhance usability and productivity, while cross-platform compatibility will enable access from various devices. Real-time analytics capabilities will provide users with up-to-date insights, and automation features will streamline repetitive tasks. Integration with third-party systems will enhance functionality and interoperability, while collaboration tools will facilitate teamwork and communication. Robust data management capabilities will ensure efficient handling of large datasets, and customization options will empower users to tailor workflows and reports. The system will be deployed on a scalable infrastructure to ensure reliability under heavy loads, and mobile access.

SYSTEM DESIGN AND DEVELOPMENT

3.1 INPUT DESIGN:

Input design for car price prediction is a critical aspect of developing a robust and effective predictive analytics system within the automotive industry. The input design determines how data is collected, organized, and prepared for analysis, ultimately influencing the accuracy and reliability of the predictive models. In the context of car price prediction, the input design encompasses various components, including data sources, data collection methods, feature selection, data preprocessing, and data integration.Data sources serve as the foundation of the input design, providing the raw material for predictive modeling. These sources may include internal databases, third-party datasets, online repositories, and real-time data streams. Internal databases contain historical information about car attributes, sales transactions, customer demographics, and market trends. Third-party datasets offer additional insights into economic indicators, consumer behavior, and regulatory factors that may impact car prices. Online repositories provide access to publicly available data such as car reviews, specifications, and pricing information. Real-time data streams deliver up-to-date information about market conditions, inventory levels, and competitor pricing strategies.Data collection methods determine how data is gathered from various sources and stored for analysis.

This may involve automated processes for extracting data from databases, web scraping techniques for collecting data from online sources, and manual data entry for capturing information from physical documents or forms. Data collection methods should be designed to ensure the integrity, completeness, and accuracy of the data while minimizing errors and biases. Feature selection is a crucial step in input design, as it involves identifying the most relevant variables or attributes that influence car prices.

3.2 OUTPUT DESIGN:

The output design for a car price prediction system is a critical aspect that directly influences user experience and decision-making processes. This design encompasses the presentation and visualization of predictive insights, recommendations, and relevant information generated by the system. By conveying information effectively and intuitively, the output design plays a crucial role in facilitating informed decision-making and maximizing the utility of the predictive analytics solution. At its core, the output design aims to present complex data and insights in a clear, concise, and visually appealing manner. It seeks to empower users with actionable information, enabling them to make informed decisions regarding car pricing, inventory management, and sales strategies. The design should cater to the diverse needs and preferences of stakeholders across the automotive value chain, including car manufacturers, dealerships, online platforms, and end consumers. One of the key components of the output design is the visualization of predictive insights and trends.

This includes graphical representations such as charts, graphs, and heat maps that illustrate patterns, correlations, and fluctuations in car prices over time. For example, line charts can be used to visualize trends in average car prices by make or model, while scatter plots can highlight relationships between price and various attributes such as mileage, year of manufacture, or geographic location. In addition to visualizations, the output design should incorporate descriptive text and contextual information to provide insights and explanations for the generated predictions.

3.3 DATABASE DESIGN:

Database design for a car price prediction system involves structuring and organizing data in a way that facilitates efficient storage, retrieval, and manipulation of information. This design process encompasses the creation of database schemas, tables, relationships, indexes, and constraints to ensure data integrity, scalability, and performance. By carefully designing the database architecture, the system can effectively manage vast amounts of data while supporting complex analytical queries and predictive modeling tasks. The first step in database design is to analyze the requirements of the car price prediction system and identify the types of data that need to be stored and managed. This includes both structured data, such as car attributes (make, model, year, mileage, price), customer information, and transactional data, as well as unstructured data, such as text from customer reviews or market trends. Understanding the nature and volume of data helps in determining the appropriate database management system (DBMS) and designing an efficient database schema. The next stage involves conceptual modeling, where the relationships between different entities and attributes are defined. This often involves techniques such as entity-relationship modeling (ER modeling), where entities represent real-world objects (e.g., cars, customers) and relationships define how these entities are connected. For example, there may be a relationship between cars and customers representing past purchases or inquiries. Based on the conceptual model, the database schema is designed, specifying the structure of tables, attributes, and relationships.

3.4 SYSTEM DEVELOPMENT:

Allocate resources, and communicate with stakeholders to ensure alignment with project goals and objectives. Agile methodologies such as System development is a comprehensive process that encompasses the design, implementation, testing, deployment, and maintenance of software solutions to address specific needs or requirements. It involves a interrelated activities and stages, each aimed at achieving the overarching goal of delivering a high-quality, reliable, and scalable system that meets the needs of its users. From conceptualization to execution, system development requires careful planning, collaboration, and iteration to ensure successThe first phase of system development is typically the requirements analysis and gathering stage. During this phase, stakeholders identify and document the functional and non-functional requirements of the system, including user needs, business objectives, and technical constraints. This involves conducting interviews, surveys, and workshops to gather input from end users, domain experts, and other relevant parties.

The goal is to establish a clear understanding of the problem domain and define the scope of the system to be developed. Once the requirements are gathered, the next phase involves system design and architecture. This phase focuses on translating the requirements into a comprehensive system architecture, encompassing the overall structure, components, interfaces, and data flows of the system. Design decisions are made regarding the selection of technologies, frameworks, and platforms to be used, as well as the division of labor among development teams. The goal is to create a blueprint for the system

3.4 DISCRIPTING MODULES:

Module 1: Import Libraries

Module 2: Create the Regressor

Module 3: Fit the Model

Module 4: Descriptive Modules

Module 5: Evaluate the Model

IMPORT LIBRARIES:

In the descriptive modules of a decision tree regressor, importing libraries is crucial for visualizing the tree structure and evaluating the model's performance. For visualization, we typically utilize Matplotlib to plot the decision tree. Additionally, for evaluating the model, we import functions to calculate metrics such as mean squared error (MSE) and mean absolute error (MAE) from the scikit-learn library.

CREATE THE REGRESSOR:

Creating a decision tree regressor involves several steps, including importing the necessary libraries, loading the dataset, preprocessing the data, splitting it into training and testing sets, training the model, and evaluating its performance. Below is a detailed explanation of each step encapsulated within a descriptive module:

FIT THE MODEL:

In the process of fitting the decision tree regressor model, we utilize the training data to train the model and enable it to learn patterns and relationships between features and the target variable. This step involves calling the fit method on the regressor object and passing the training data as arguments. Here's how we encapsulate this process within a descriptive module:

DESCRIPTIVE MODELS: Descriptive modules in the context of machine learning typically refer to the components or steps involved in understanding and interpreting the model, its behavior, and the underlying data. In the case of a decision tree regressor, descriptive modules.

ALGORITHM

DECISION TREE REGRESSOR:

Decision Tree Regressor is a powerful machine learning algorithm used extensively in predictive modeling, including car price prediction. At its essence, a Decision Tree Regressor is a tree-like model where each node represents a feature or attribute, each branch represents a decision based on that feature, and each leaf node represents the outcome or prediction. Unlike classification trees that predict categorical labels, regression trees predict continuous numerical values, making them well-suited for tasks like predicting car prices based on various attributes. The construction of a Decision Tree Regressor begins with the selection of the best feature tot the data into subsets that maximize the homogeneity of the target variable, in this case, the car price. This process is repeated recursively for each subset until a stopping criterion is met, such as reaching a maximum tree depth, achieving minimum sample size, or when further splitting does not significantly improve predictive performance.

The resulting tree represents a hierarchical set of decision rules that can be used to make predictions for new instances of the key advantages of Decision Tree Regressors is their interpretability. The resulting tree structure is easy to visualize and understand, making it possible to interpret the decision-making process and understand which features are most influential in predicting car prices. This transparency is valuable for stakeholders who seek to gain insights into the factors driving car prices and understand how different attributes contribute to the final prediction. Additionally, Decision Tree Regressors are versatile and can handle both numerical and categorical features without the need for extensive data preprocessing. This flexibility makes them well-suited for datasets with mixed data types, such as those commonly encountered in car price prediction tasks. Furthermore, Decision Tree Regressors can automatically handle missing values by choosing the best split based on available data, reducing the need for imputation techniques. However, Decision Tree Regressors are prone to over fitting, especially when the tree depth is not appropriately constrained. Over fitting occurs when the model captures noise or outliers in the training data, leading to poor generalization performance on unseen data. To mitigate over fitting, techniques such as pruning, limiting the maximum tree depth, or setting minimum samples per leaf node are commonly employed. These regularization techniques help prevent the model from becoming overly complex and ensure that it generalizes well to unseen data.

TESTING AND IMPLEMENTATION

4.1 TESTING

UNIT TESTING

Unit testing for car price prediction involves breaking down the prediction process into individual components and testing each component separately to ensure they function as expected. This typically includes defining various test cases covering different input scenarios, such as different car features and edge cases. Using a unit testing framework like, you can implement these test cases, execute them, and verify that the actual outputs match the expected outputs. This iterative process helps ensure the accuracy and reliability of the car price prediction software.

INTEGRATED TESTING

Integrated testing for car price prediction involves testing the interaction between different modules or components of the software system that collectively contribute to the prediction process. This testing ensures that the integration of these components produces the desired output.

SYSTEM TESTING

System testing for car price prediction involves evaluating the entire software system as a whole to ensure it meets the specified requirements and performs as expected in its intended environment. This testing phase focuses on validating the system's functionality, performance, reliability, and usability

ACCEPATANCE TESTING

Acceptance testing for car price prediction involves verifying that the software meets the requirements and expectations of the end users or stakeholders. This testing phase typically occurs towards the end of the development cycle and involves testing the system in a real-world or simulated environment. In the context of car price prediction, acceptance testing would entail validating the software against predetermined criteria, such as accuracy, ease of use, and compatibility with existing systems

4.2 IMPLEMENTATION

Implementing a car price prediction system involves translating the conceptual design and predictive models into functioning software that can effectively analyze data, generate predictions, and provide insights to users. This implementation process encompasses several stages, including data preprocessing, model development, integration, testing, deployment, and maintenance, each of which requires careful planning, development, and validation to ensure the system's effectiveness and reliability.The implementation process begins with data preprocessing, where raw data on car attributes, historical sales records, market trends, and other relevant information are collected, cleaned, and transformed into a format suitable for analysis. This involves handling missing values, removing outliers, encoding categorical variables, and scaling numerical features to ensure consistency and quality in the data. Data preprocessing techniques such as feature engineering, dimensionality reduction, and normalization are applied to prepare the data for model development.

Once the data is prepared, the next step is to develop predictive models for car price prediction. Various machine learning algorithms and techniques, such as decision trees, random forests, support vector machines, gradient boosting, and neural networks, may be employed to develop predictive models based on the prepared data. Model selection, hyperparameter tuning, and cross-validation techniques are used to optimize the performance of the models and ensure their robustness and generalization ability.After developing the predictive models, the next phase is integration, where the models are integrated into the software system along with other components such as user interfaces, data storage, and external APIs.

5. CONCLUSION

The grand tapestry of human existence, woven from the threads of history, culture, and innovation, lies an intricate web of interconnectedness that transcends the boundaries of time and space. From the earliest civilizations that emerged along the fertile banks of ancient rivers to the bustling metropolises of the modern era, humanity has embarked on a relentless journey of discovery and progress, driven by an insatiable curiosity and an innate desire to unravel the mysteries of the universe.At the heart of this ceaseless quest lies the pursuit of knowledge, the cornerstone upon which civilizations have risen and fallen, and the beacon that guides humanity towards enlightenment and understanding. Through the ages, philosophers, scholars, and visionaries have sought to unlock the secrets of the cosmos, delving into the realms of science, literature, and philosophy in search of truth and wisdom. From the groundbreaking theories of Aristotle and Plato to the revolutionary discoveries of Galileo and Newton, the annals of history are replete with tales of intellectual triumph and scientific innovation that have reshaped the very fabric of human existence. With each passing era, new horizons of knowledge have been unveiled, unveiling the boundless potential of the human mind and igniting the flames of progress and innovation that have propelled civilization forward.Yet, amidst the vast expanse of human achievement and endeavor, there exists a paradoxical dichotomy, a duality of light and shadow that casts its long shadow across the annals of history. For as humanity has ascended to the lofty heights of intellectual prowess and technological mastery, so too has it grappled with the darker impulses that lurk within the recesses of the human psyche, giving rise to conflict, oppression, and suffering on an unprecedented scale.

6. BIBLIOGRAGHY

6.1 BOOK REFERRENCES

Building Machine Learning Powered Applications: Going from Idea to Product" by Emmanuel Ameisen

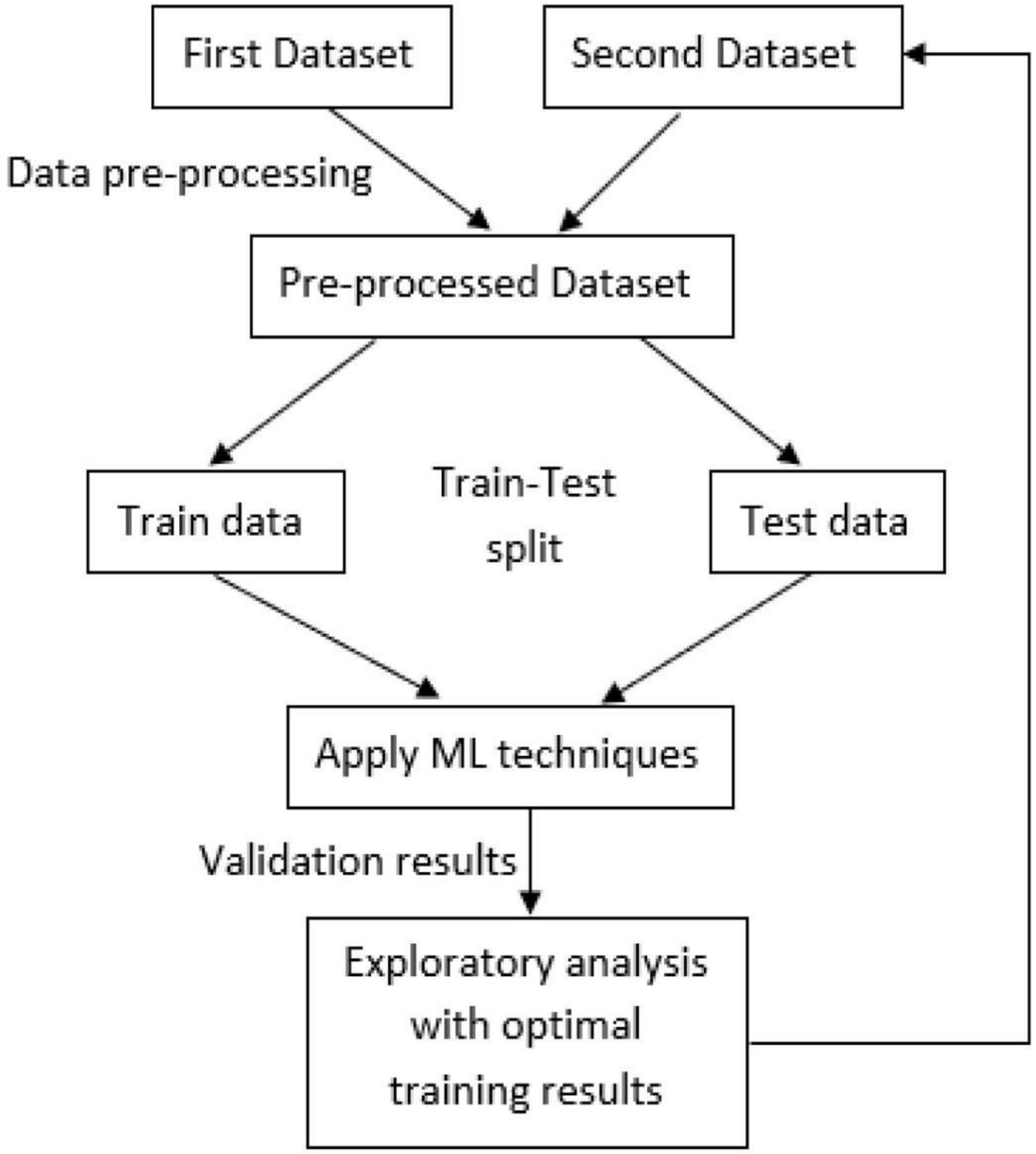
"Predictive Analytics: The Power to Predict Who Will Click, Buy, Lie, or Die" by Eric Siegel

6.2 WEBSITES REFERRENCES

https://www.semanticscholar.org/paper/Car-Price-Prediction-using-Machine-Learning-Gegic-Isakovic/8b101952d39ad3d33712e2b92ab3c6c19295ca3c

7 .APPENDICES

A.DATA FLOWDIAGRAM



B. SCREEN SHOT

In [2]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeRegressor

data = pd.read_csv("CarPrice.csv")
data.head()
```

Out[2]:

	car_ID	symboling	CarName	fueltype	aspiration	doornumber	carbody	drivewheel	engi
0	1	3	alfa-romero giulia	gas	std	two	convertible	rwd	
1	2	3	alfa-romero stelvio	gas	std	two	convertible	rwd	
2	3	1	alfa-romero Quadrifoglio	gas	std	two	hatchback	rwd	
3	4	2	audi 100 ls	gas	std	four	sedan	fwd	
4	5	2	audi 100ls	gas	std	four	sedan	4wd	

5 rows × 26 columns

In [ ]:

In [3]:
```
data.isnull().sum()
```

Out[3]:
```
car_ID              0
symboling           0
CarName             0
fueltype            0
aspiration          0
doornumber          0
carbody             0
drivewheel          0
enginelocation      0
wheelbase           0
carlength           0
carwidth            0
carheight           0
curbweight          0
enginetype          0
cylindernumber      0
enginesize          0
fuelsystem          0
boreratio           0
stroke              0
compressionratio    0
horsepower          0
peakrpm             0
citympg             0
highwaympg          0
price               0
dtype: int64
```

In [4]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 205 entries, 0 to 204
Data columns (total 26 columns):
 #   Column            Non-Null Count  Dtype  
---  ------            --------------  -----  
 0   car_ID            205 non-null    int64  
 1   symboling         205 non-null    int64  
 2   CarName           205 non-null    object 
 3   fueltype          205 non-null    object 
 4   aspiration        205 non-null    object 
 5   doornumber        205 non-null    object 
 6   carbody           205 non-null    object 
 7   drivewheel        205 non-null    object 
 8   enginelocation    205 non-null    object 
 9   wheelbase         205 non-null    float64
 10  carlength         205 non-null    float64
 11  carwidth          205 non-null    float64
 12  carheight         205 non-null    float64
 13  curbweight        205 non-null    int64  
 14  enginetype        205 non-null    object 
 15  cylindernumber    205 non-null    object 
 16  enginesize        205 non-null    int64  
 17  fuelsystem        205 non-null    object 
 18  boreratio         205 non-null    float64
 19  stroke            205 non-null    float64
 20  compressionratio  205 non-null    float64
 21  horsepower        205 non-null    int64  
 22  peakrpm           205 non-null    int64  
 23  citympg           205 non-null    int64  
 24  highwaympg        205 non-null    int64  
 25  price             205 non-null    float64
dtypes: float64(8), int64(8), object(10)
memory usage: 41.8+ KB
```

In [5]:
```
print(data.describe())
```

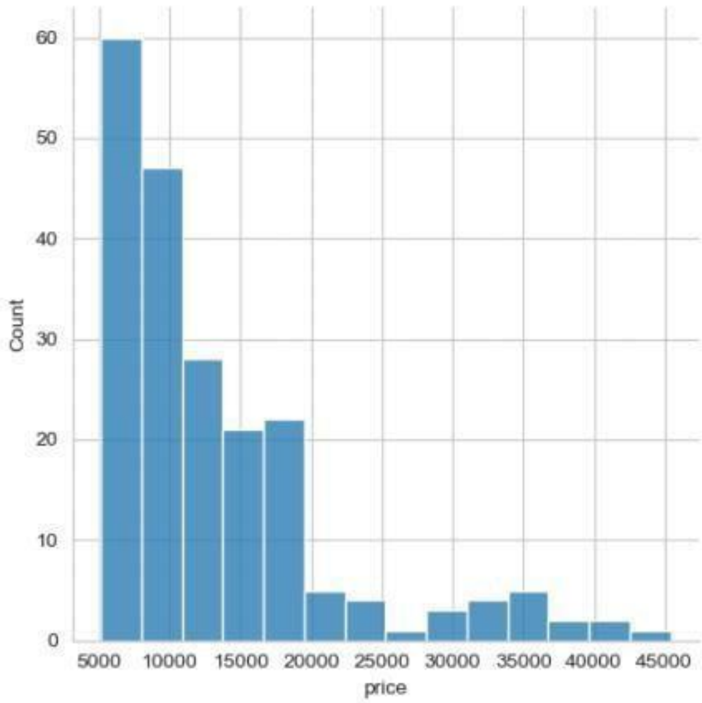
```
           car_ID   symboling   wheelbase   carlength    carwidth   carheight  \
count  205.000000  205.000000  205.000000  205.000000  205.000000  205.000000
mean   103.000000    0.834146   98.756585  174.049268   65.907805   53.724878
std     59.322565    1.245307    6.021776   12.337289    2.145204    2.443522
min      1.000000   -2.000000   86.600000  141.100000   60.300000   47.800000
25%     52.000000    0.000000   94.500000  166.300000   64.100000   52.000000
50%    103.000000    1.000000   97.000000  173.200000   65.500000   54.100000
75%    154.000000    2.000000  102.400000  183.100000   66.900000   55.500000
max    205.000000    3.000000  120.900000  208.100000   72.300000   59.800000

        curbweight  enginesize   boreratio      stroke  compressionratio  \
count   205.000000  205.000000  205.000000  205.000000        205.000000
mean   2555.565854  126.907317    3.329756    3.255415         10.142537
std     520.680204   41.642693    0.270844    0.313597          3.972040
min    1488.000000   61.000000    2.540000    2.070000          7.000000
25%    2145.000000   97.000000    3.150000    3.110000          8.600000
50%    2414.000000  120.000000    3.310000    3.290000          9.000000
75%    2935.000000  141.000000    3.580000    3.410000          9.400000
max    4066.000000  326.000000    3.940000    4.170000         23.000000

       horsepower      peakrpm     citympg  highwaympg         price
count  205.000000   205.000000  205.000000  205.000000    205.000000
mean   104.117073  5125.121951   25.219512   30.751220  13276.710571
std     39.544167   476.985643    6.542142    6.886443   7988.852332
min     48.000000  4150.000000   13.000000   16.000000   5118.000000
25%     70.000000  4800.000000   19.000000   25.000000   7788.000000
50%     95.000000  5200.000000   24.000000   30.000000  10295.000000
75%    116.000000  5500.000000   30.000000   34.000000  16503.000000
max    288.000000  6600.000000   49.000000   54.000000  45400.000000
```

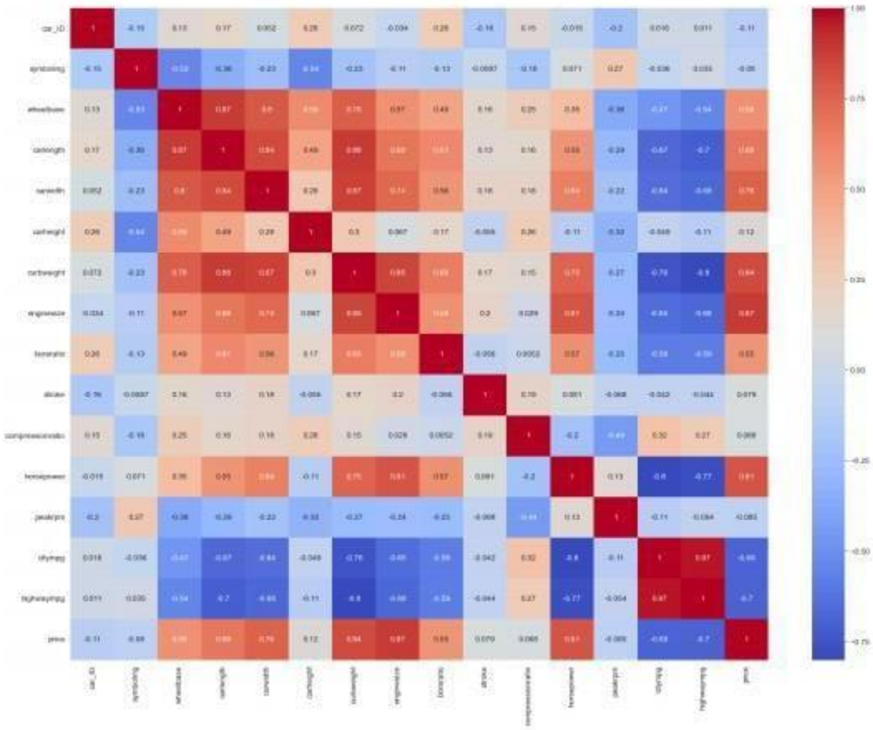
In [7]:
```
sns.set_style("whitegrid")
plt.figure(figsize=(15, 10))
sns.displot(data.price)
plt.show()
```

<Figure size 1500x1000 with 0 Axes>



In [10]:
```
plt.figure(figsize=(20, 15))
correlations = data.corr()
sns.heatmap(correlations,cmap="coolwarm",annot=True)
plt.show()
```

```
C:\Users\Student\AppData\Local\Temp\ipykernel_5924\3725037003.py:2: FutureWar
ning: The default value of numeric_only in DataFrame.corr is deprecated. In a
future version, it will default to False. Select only valid columns or specif
y the value of numeric_only to silence this warning.
  correlations = data.corr()
```



In [11]:
```
data.CarName.unique()
```

Out[11]:
```
array(['alfa-romero giulia', 'alfa-romero stelvio',
       'alfa-romero Quadrifoglio', 'audi 100 ls', 'audi 100ls',
       'audi fox', 'audi 5000', 'audi 4000', 'audi 5000s (diesel)',
       'bmw 320i', 'bmw x1', 'bmw x3', 'bmw z4', 'bmw x4', 'bmw x5',
       'chevrolet impala', 'chevrolet monte carlo', 'chevrolet vega 2300',
       'dodge rampage', 'dodge challenger se', 'dodge d200',
       'dodge monaco (sw)', 'dodge colt hardtop', 'dodge colt (sw)',
       'dodge coronet custom', 'dodge dart custom',
       'dodge coronet custom (sw)', 'honda civic', 'honda civic cvcc',
       'honda accord cvcc', 'honda accord lx', 'honda civic 1500 gl',
       'honda accord', 'honda civic 1300', 'honda prelude',
       'honda civic (auto)', 'isuzu MU-X', 'isuzu D-Max ',
       'isuzu D-Max V-Cross', 'jaguar xj', 'jaguar xf', 'jaguar xk',
       'maxda rx3', 'maxda glc deluxe', 'mazda rx2 coupe', 'mazda rx-4',
       'mazda glc deluxe', 'mazda 626', 'mazda glc', 'mazda rx-7 gs',
       'mazda glc 4', 'mazda glc custom l', 'mazda glc custom',
       'buick electra 225 custom', 'buick century luxus (sw)',
       'buick century', 'buick skyhawk', 'buick opel isuzu deluxe',
       'buick skylark', 'buick century special',
       'buick regal sport coupe (turbo)', 'mercury cougar',
       'mitsubishi mirage', 'mitsubishi lancer', 'mitsubishi outlander',
       'mitsubishi g4', 'mitsubishi mirage g4', 'mitsubishi montero',
       'mitsubishi pajero', 'Nissan versa', 'nissan gt-r', 'nissan rogue',
       'nissan latio', 'nissan titan', 'nissan leaf', 'nissan juke',
       'nissan note', 'nissan clipper', 'nissan nv200', 'nissan dayz',
       'nissan fuga', 'nissan otti', 'nissan teana', 'nissan kicks',
       'peugeot 504', 'peugeot 304', 'peugeot 504 (sw)', 'peugeot 604sl',
       'peugeot 505s turbo diesel', 'plymouth fury iii',
       'plymouth cricket', 'plymouth satellite custom (sw)',
       'plymouth fury gran sedan', 'plymouth valiant', 'plymouth duster',
       'porsche macan', 'porcshce panamera', 'porsche cayenne',
       'porsche boxter', 'renault 12tl', 'renault 5 gtl', 'saab 99e',
       'saab 99le', 'saab 99gle', 'subaru', 'subaru dl', 'subaru brz',
       'subaru baja', 'subaru r1', 'subaru r2', 'subaru trezia',
       'subaru tribeca', 'toyota corona mark ii', 'toyota corona',
       'toyota corolla 1200', 'toyota corona hardtop',
       'toyota corolla 1600 (sw)', 'toyota carina', 'toyota mark ii',
       'toyota corolla', 'toyota corolla liftback',
       'toyota celica gt liftback', 'toyota corolla tercel',
       'toyota corona liftback', 'toyota starlet', 'toyota tercel',
       'toyota cressida', 'toyota celica gt', 'toyouta tercel',
       'vokswagen rabbit', 'volkswagen 1131 deluxe sedan',
       'volkswagen model 111', 'volkswagen type 3', 'volkswagen 411 (sw)',
       'volkswagen super beetle', 'volkswagen dasher', 'vw dasher',
       'vw rabbit', 'volkswagen rabbit', 'volkswagen rabbit custom',
       'volvo 145e (sw)', 'volvo 144ea', 'volvo 244dl', 'volvo 245',
       'volvo 264gl', 'volvo diesel', 'volvo 246'], dtype=object)
```

In [ ]:

C. CODING

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import sea born as sns
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeRegressor
data = pd.read_csv("CarPrice.csv")
data.head
data.isnull().sum()
data.info()
print(data.describe())
sns.set_style("whitegrid")
plt.figure(figsize=(15, 10))
sns.distplot(data.price)
plt.show()
```