**MD2201 Data Science**

**Course Project**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **S.No** | **Div** | **Batch No** | **Group No** | **Roll No** | **Gr.No** | **Name of Student** |
| **1** | **IT-B** | **1** | **1** | **2** | **12211665** | **Pranav Kale** |
| **2** | **13** | **12210893** | **Maitrey Katkar** |
| **3** | **17** | **12210313** | **Omkar Khanvilkar** |
| **4** | **18** | **12210610** | **Janvi Kharat** |
| **5** | **20** | **12210505** | **Gita Kolate** |
| **6** | **21** | **12210697** | **Babusha Kolhe** |

**Science**

**Date of performance:**

# Project Title: OLX Car Price Prediction

# Data Set Name: Used Cars Price Prediction

# Data Set Source: Kaggle

# Data set Link: <https://kaggle.com/datasets/avikasliwal/used-cars-price-prediction?select=train-data.csv>

# Data Set Description:

# *This Dataset consist features or columns like:*

· **Name:** full name of the cars

· **Location:** Location of the car owner

· **Year:** Launch Year of particular car model

· **Kilometers\_Driven:** Kilometres Driven by particular car

· **Fuel\_Type:** Fuel Type of the car (Petrol, Diesel, Hybrid)

· **Transmission:** Type of transmission (manual, auto)

· **Owner\_Type:** Which type of owner (first, second, third)

· **Mileage:** Mileage of the car

· **Engine:** Engine capacity of the car

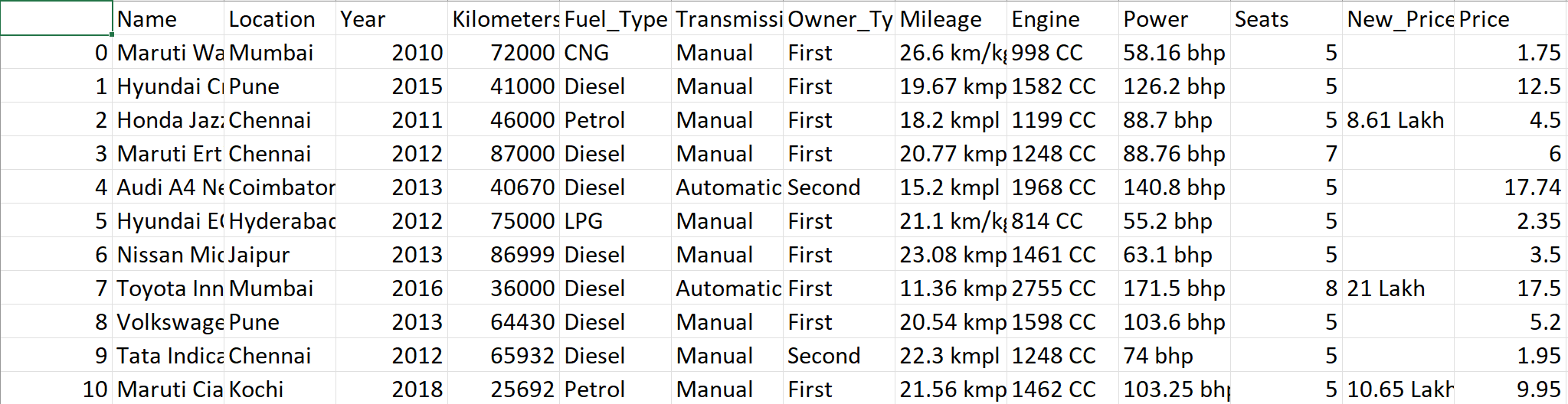
· **Power:** Power of the car

· **Seats:** Number of seats inside the car

· **New\_Price:** Market price of new model of same car

· **Price:** Actual Price of the car asked by the owner

**Preview of dataset**



* Containing 6019 rows and 14 columns for training and for testing there are 1234 rows and 13 columns the training and testing data are provided by the author separately.
* There are total 1876 unique cars data is available in training dataset
* This data is collected on 1998 to 2019 car models

# Description of Work Done:

**Steps Perform :**

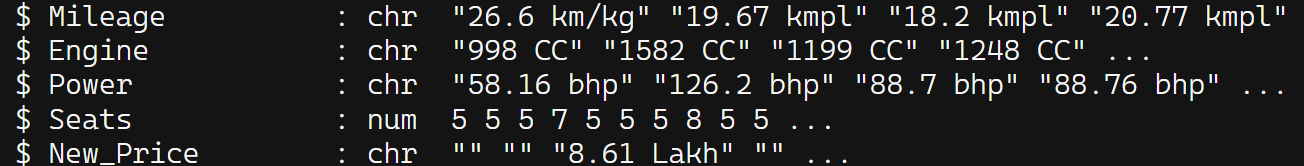
Data Accumulation 🡪 Data Preprocessing 🡪 Model Training 🡪 Model Testing 🡪 Visualization 🡪 Deployment

# Literature Survey: Give the tabular form of 20 papers with the following columns information.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| S.No | Title of paper Authors of paper | Name of journal/conference and date | Data Set name ans link | Data preprocessing techniques done(class imbalance, normalization, missing values handling etc) | Algorithms applied | Findings (Quantitative) |
| 1 | Used Car Price Prediction using K-Nearest Neighbor Based ModelK. Samruddhi Dr. R. Ashok Kumar | International Journal of Innovative Research in Applied Sciences and Engineering (IJIRASE) | https://www.kaggle.com/datasets | The primary data preprocessing technique used here is feature engineering, which involves manipulating andtransforming raw data into a format suitable for machine learning algorithms. | K-Nearest Neighbor (KNN) | Accuracy : 85% |
| 2 | Predicting the Price of Used Cars using Machine Learning TechniquesSameerchand Pudaruth | International Journal of Information & Computation Technology. | https://www.kaggle.com/datasets | NormalizationHandling missing valuesData Reduction | Multiple Linear Regression AnalysisK-Nearest Neighbors(KNN)Naive BayesDecision Trees | Mean Error: Rs51,000 |
| 3 | Prediction of The Prices of Second-Hand CarsOzer Celik, U. Omer Osmanoglu | European Journal of Science and Technology | http://ikinciyeni.com/ | **Data Collection****Data Labeling** | Linear Regression Analysis | R-squared values ranged from 0.71 to 0.92. |
| 4 | Used car price prediction using linear regression modelAshutosh Datt Sharma\*1, Vibhor Sharma\*2 | International Research Journal of Modernization in Engineering Technology and Science | http://Kaggle.com | Null-Entry RemovalOne-Hot EncodingTrain-test splitFeature Selection | Linear regression model | R 2 value of 0.86 |
| 5 | Used Cars Price Prediction and Valuation using Data Mining TechniquesAbdulla AlShared | RIT Digital Institutional Repository | Data was collected and Scrapped from a website BuyAnyCar | Handling missing valuesConverting Categorical dataHandling OutliersData Normalization | Random Forest RegressorLinear RegressionBagging Regressor | Random Forest Regressor: Accuracy: Achieved an accuracy of 95%. Mean Squared Error (MSE): 0.025. Mean Absolute Error (MAE): 0.0008. Root Mean Squared Error (RMSE): 0.03. |
| 6 | Price Prediction for Used CarsMarcus Collard | International Research Journal of Modernization in Engineering Technology and Science | <http://kaggle>.com/ | Data Cleaning and NormalizationConversion of categorical variables to numeric | Linear RegressionRidge regressionLasso RegressionRandom Forest Regression | Random Forest RegressionRMSE value of 4799MAPE value of 37.65% |
| 7 | Pre-owned car price prediction by employing machine learning techniquesMauparna Nandan Debolina Ghosh | Journal of Decision Analytics and Intelligent Computing | <https://www>.google.com/url? | Label EncoderData Normalization | Random Forest | MAE : 0.167132MSE : 0.078840RMSE : 0.078840R2 Score : 0.867691Accuracy : 86.769137 |
| 8 | Advancing Used Car Price Prediction in South Africa: An Empirical Examination of Machine Learning Techniques Zenzele Abel Msiza,Pius Adewale Owolawi | 2023 International Conference on Artificial Intelligence and its Applications | Data obtained from Demo automobiles website | Data cleaning and NormalizationHandling Outliers | Linear Regression, Decision Tree, Random Forest, Gradient Boosted Trees Regressor, Artificial Neural Network, and K-Nearest Neighbors. The Random Forest method | Random Forest (RF): R-squared value: 0.988 RMSE value: 0.019 |
| 9 | Using Linear Regression For Used Car Price PredictionSümeyra MUTİ1, Kazım YILDIZ2 | International Journal of Computational and Experimental Science and Engineering | <http://Kaggle>.com | Handling Missing or incorrect valuesHandling outliersFeature Transformation | Linear regression model | R-squared value: 0.62 |
| 10 | Used Car Price Prediction Using Machine LearningVELURU RANJITH | KARUNYA INSTITUTE OF TECHNOLOGY AND SCIENCES Karunya Nagar, Coimbatore – 641 114. INDIA | <https://www>.kaggle.com/datasets/lepchenkov/usedcarscatalog | Removing Outliers | Random Forest | Accuracy: 0.861908 |
| 11 | Prediction of Used Car Prices using Machine Learning Techniques  Eesha Pandit1, Hitanshu Parekh2, Pritam Pashte3, Aakash Natani4 | International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056 Volume: 09 Issue: 12 | Dec 2022 | <http://Kaggle>.com | Feature RenamingFeature SelectionExploratory Data Analysis (EDA)One-Hot EncodingCorrelation AnalysisFeature Allocation | Linear RegressionLasso RegressionRidge RegressionBayesian Ridge RegressionRandom Forest Regression | Random Forest RegressionR-squared (r2) score of 0.95. |
| 12 | CAR PRICE PREDICTION USING MACHINE LEARNING TECHNIQUES   Abishek R\*1 | International Research Journal of Modernization in Engineering Technology and Science | <http://Kaggle>.com | Applied machine learning techniques to clean and pre-process the dataset. Removed missing values, outliers, and irrelevant features. | 1. Simple Linear Regression2. Multiple Linear Regression3.Clustering Methods (e.g., K-means)4. Logistic Regression5. K-nearest Neighbors (KNN)6. Random Forest7. Decision Tree | f random forest modelMAE : 1.522771460587MSE : 10.49007991840RMSE : 3.2388392856711R-squared (r2) : 0.910486881527 |
| 13 | Prediction of the price of used cars based on machine learning algorithms  Yian Zhu | Proceedings of the 3rd International Conference on Signal Processing and Machine Learning | https://tianchi.aliyun.com/dataset/?lang=en-us | Data cleaning- missing values- outliers- duplicate valuesData dimension reductionLinear – pcaNon linear -ISOMAPFeature selections | XGBoostSVMNeural network | R-squared (r2) score of 0.9823. |
| 14 | Used Cars Price Prediction using Supervised Learning Techniques  Pattabiraman Venkatasubbu, Mukkesh Ganesh | International Journal of Engineering and Advanced Technology (IJEAT) | . The data was collected from the 2005 Central Edition of the Kelly Blue Book | Applied machine learning techniques to clean and pre-process the dataset. Removed missing values, outliers, and irrelevant features. | Lasso RegressionMultiple RegressionRegression Tree | Error rate:Multiple regression:3.468 % |
| 15 | Price Prediction of Used Cars Using Linear Regression  1Amit Kewat, 2Nitesh Kanojiya | Journal of Online Engineering Education | https://www.kaggle.com/datasets | Data Cleaning and NormalizationHandling OutliersExtracting numeric values | Linear Regression | Accuracy : 89% |
| 16 | Car Price Prediction using Machine Learning Techniques Enis Gegic, Becir Isakovic, Dino Keco, Zerina Masetic, Jasmin Kevric | TEM Journal | autopijaca.ba | Data Cleaning Skewed Class RemovalNormalization Conversion of Continuous Attributes into Categorical Values Conversion of Regression Prediction Problem into Classification Problem | Random Forest (RF) Classifier Artificial Neural Network (ANN) Classifier Support Vector Machine (SVM) Classifier | For the Cheap subset, SVM achieved the highest accuracy at 86.96%. For the Moderate subset, ANN performed better with an accuracy of 86.11%. For the Expensive subset, SVM achieved the highest accuracy at 90.48%. |
| 17 | Used Car Price Prediction Using Random Forest Algorithm  Prof. Dipti A. Gaikwad1 , Pratik S. Suwarnakar2 , Yash R. Mahajan3 , Amita U. Petkar4 , Shreyasi G. Theurkar5 | International Journal for Multidisciplinary Research (IJFMR) | https://www.kaggle.com/datasets | Data CleaningFeature EngineeringNormalization/StandardizationOne-Hot EncodingTrain-Test Split | Linear Regression Lasso Regression Support Vector Machine (SVM) Random Forest | Random ForestR2 Score: 0.8697 |
| 18 | Used Car Price Prediction Using Machine Learning Techniques  Mrs Shyamali Das1 , Mr Ananta Laha2 , Mr Alok Jena3 , Ms Priyadarshini Samal4 | International Journal of Research Publication and Reviews | https://www.kaggle.com/datasets | 1. Data Cleaning 2. Encoding Categorical Variables 3. Feature Scaling 4. Feature Engineering 5. Handling Skewed Data 6. Train-Test Split | Linear Regression Lasso Regression Support Vector Machine (SVM) Random Forest | Accuracy by Implementation of Random Forest 91. 435 % |
| 19 | Used car price prediction  Abhishek Jha, Dr. Ramveer Singh, Manish, Imran Saifi, Shipra Srivastava | International Journal of Advance Research, Ideas and Innovations in Technology | cardekho.com | Removing Outliers | Random Forest Regression | Accuracy : 91.45% |
| 20 | Prediction of prices for used car by using regression models  [Nitis Monburinon](https://www.semanticscholar.org/author/Nitis-Monburinon/50845488), [Prajak Chertchom](https://www.semanticscholar.org/author/Prajak-Chertchom/50843855), [T. Kaewkiriya](https://www.semanticscholar.org/author/T.-Kaewkiriya/1718286), [Suwat Rungpheung](https://www.semanticscholar.org/author/Suwat-Rungpheung/50844095), [Sabir Buya](https://www.semanticscholar.org/author/Sabir-Buya/50847375), [Pitchayakit Boonpou](https://www.semanticscholar.org/author/Pitchayakit-Boonpou/50844632) | 2018 5th International Conference on Business and Industrial Research (ICBIR) | https://www.kaggle.com/datasets | 1. Data Cleaning 2. Encoding Categorical Variables 3. Feature Scaling 4. Feature Engineering 5. Handling Skewed Data 6. Train-Test Split | Gradient Boosted Regression TreesRandom Forest RegressionMultiple Linear Regression | Gradient Boosted Regression TreesMSE : 0.28  highest accuracy |

# Data Preprocessing (if any):

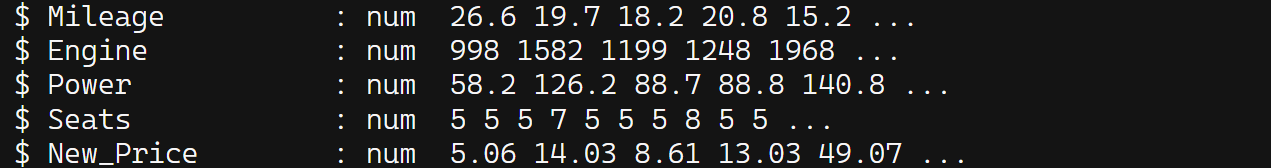
* While observing the structure of the dataset we observed datatypes of some variables or columns are not suitable for our further processes and also some columns contain NA values.
* And some string type columns contain empty strings. Like New\_Price, Mileage, Engine, Power etc
* There where total 42 rows containing “Na” values.
* And there are some columns which should be in datatype Integer or Double but they are present in dataset as a Character or String.

Columns -> Mileage, Engine, Power, New\_Price

* And above columns also contain some units in suffix which is not very useful for use in further process.
* And the New\_Price column consists lots of Empty strings and so after converting it into integer or float the empty string denoted by “Na” and due to high number of Na values in this column we should impute this column.

So above are the some impurities or problems in our dataset and we clean and solve those as follows:

1. We first delete or remove the 42 rows from the dataset which contain “Na” values at start.
2. After that we convert all string type columns like Mileage, Engine, Power into double and also remove the suffix from it.
3. And also Imputed the New\_Price values and converted into double datatype Imputation is done using rfImpute() function present in randomforest library uses Proximity Matrix concept to impute values



1. Now our data is ready for further process after data preprocessing there are total 5977 rows are remain in the training data.
2. Similarly we perform same data preprocessing on test data and there are 1192 rows are remain in the testing data.

# Feature Selection (if any): *Explain the different feature selection techniques you have used in the project*

# Algorithms Implemented:

* As our problem statement is of Regression type so we are using following models :

1. Multiple Linear Regression
2. Decision Tree
3. Random Forest
4. SVM (Support Vector Machine as Regressor)

* Before training each model we perform some common processes for better results and accuracy

1. As our car's names are too long which causes problems while training since we consider them as factors or categories so we first make it short till 2 String.
2. And as we reduce the car name string we add one more extra feature into the data which is “Brand” which denotes the brand of the car and we found there are 32 different Brands of Cars present in the entire dataset.
3. We convert some columns like Name, Brand, Owner\_Type, Transmission, Location etc into a factor or category.
4. And first separate out the price column from the testing dataset and as the dataset is already divided into 75:25 ratio for training and testing respectively.
5. And after all of this we send the training data to a different model.

* **Multiple Linear Regression:**

Here we use “lm()” function to train the linear model

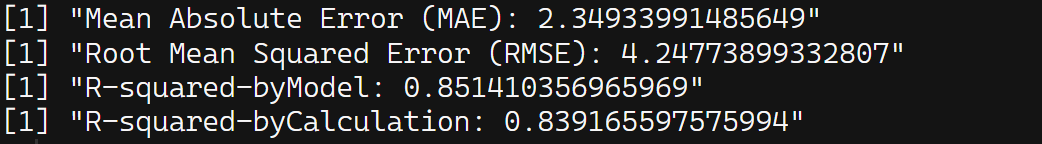


Here we train our model on all columns for price prediction.

And after training of Model we predict the price of cars present in test data.



After Prediction we test model Accuracy using R^2, RMSE & MAE values.



* **Decision Tree**

Here we use “rpart()” function to train the decision tree model



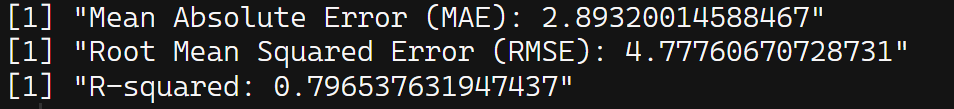
Here we train our model on all columns for price prediction.

The method specified as "anova" indicates that the decision tree will employ analysis of variance to determine the splits at each node during the tree-building process.

And after training of Model we predict the price of cars present in test data.



After Prediction we test model Accuracy using R^2, RMSE & MAE values



* **Random Forest**

Here we use “randomForest()” function to train the random forest model



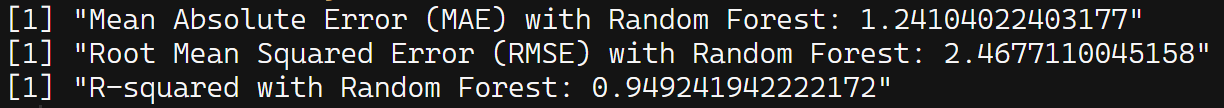
Here we train our model on all columns for price prediction.

The parameter iter = 300 specifies the number of trees to be grown in the random forest ensemble, with 300 trees being grown in this case.

And after training of Model we predict the price of cars present in test data.



After Prediction we test model Accuracy using R^2, RMSE & MAE values



* **SVM**

Here we use “svm()” function to train the SVM model



Here we train our model on all columns for price prediction.

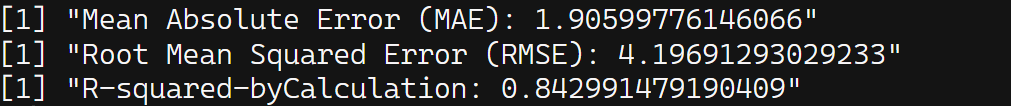
The parameter na.action = na.omit specifies the action to take if there are missing values in the data, instructing the model to omit observations with missing values. The parameter scale = TRUE indicates that the predictors should be scaled to have zero mean and unit variance, which is a common preprocessing step in SVM models to ensure that all features are on a similar scale

The kernel function used in this SVM model is specified as 'radial', which denotes a radial basis function (RBF) kernel. RBF kernels are commonly used in SVM models for non-linear regression tasks as they can effectively capture complex relationships between predictors and the target variable.

And after training of Model, we predict the price of cars present in test data.



After Prediction, we test model Accuracy using R^2, RMSE & MAE values



|  |  |  |  |
| --- | --- | --- | --- |
| Model | MAE | RMSE | R-Squared |
| Multiple Linear Regression | 2.3493399 | 4.2477389 | 0.85141035 |
| Decision Tree | 2.920059 | 4.766653 | 0.8014424 |
| Random Forest | 1.24104022 | 2.46771 | 0.94924194 |
| SVM (radial-kernel) | 1.9045498 | 4.1931021 | 0.843276 |

# After Checking Significance of each X-variables we get to know that “Seats” variable have least significance that of others

# 

# But After removing “Seats” the accuracy or R^2 value decreases

|  |  |  |  |
| --- | --- | --- | --- |
| Model | MAE | RMSE | R-Squared |
| Multiple Linear Regression | 2.87381 | 4.97619 | 0.766461 |

# Code*:*

# Model Training code:

library(caret)

library(randomForest)

library(dplyr)

data<-read.csv("Final\_TrainingDataSet.csv")

*# Remove unnecessary columns*

data <- data[, !(names(data) %in% c("New\_Price", "X","Year"))]

*# Extract only the first string from the Name column*

*# data$Name <- sapply(strsplit(data$Name, " "), function(x) paste(x[1:min(length(x), 2)], collapse=" "))*

data$Brand <- sapply(strsplit(data$Name, " "), function(x) paste(x[1:min(length(x), 1)], collapse=" "))

data <- data[, !(names(data) %in% c("Name","X.1"))]

*# Convert required columns to factor*

*# This column has more than 53 levels or categories*

data$Location <- as.factor(data$Location)

data$Fuel\_Type <- as.factor(data$Fuel\_Type)

data$Transmission <- as.factor(data$Transmission)

data$Owner\_Type <- as.factor(data$Owner\_Type)

data$Brand<- as.factor(data$Brand)

*# Divide dataset into training and testing (75% train, 25% test)*

set.seed(123) *# for reproducibility*

pd <- sample(2, nrow(data), replace = TRUE, prob = c(0.75,0.25))

train\_data <- data[pd == 1, ]

test\_data <- data[pd == 2, ]

write.csv(train\_data, file = "train\_data.csv")

Price = test\_data$Price

test\_data <- test\_data[, !(names(data) %in% c("Price"))]

*# Convert categorical variables to factors with levels from training data*

test\_data$Location <- factor(test\_data$Location, levels = levels(train\_data$Location))

test\_data$Fuel\_Type <- factor(test\_data$Fuel\_Type, levels = levels(train\_data$Fuel\_Type))

test\_data$Transmission <- factor(test\_data$Transmission, levels = levels(train\_data$Transmission))

test\_data$Owner\_Type <- factor(test\_data$Owner\_Type, levels = levels(train\_data$Owner\_Type))

test\_data$Brand <- factor(test\_data$Brand, levels = levels(train\_data$Brand))

*# names(test\_data)*

*# is.numeric(test\_data$Year)*

*# is.numeric(test\_data$Age)*

*# is.numeric(test\_data$Kilometers\_Driven)*

*# is.numeric(test\_data$Mileage)*

*# is.numeric(test\_data$Engine)*

*# is.numeric(test\_data$Power)*

*# is.numeric(test\_data$Seats)*

*# is.factor(test\_data$Location)*

*# is.factor(test\_data$Fuel\_Type)*

*# is.factor(test\_data$Transmission)*

*# is.factor(test\_data$Owner\_Type)*

*# is.factor(test\_data$Brand)*

*# Train a Random Forest model for price prediction*

rf\_model <- randomForest(Price ~ ., data = train\_data,iter=300)

*# Predict price using test data*

predicted\_price\_rf <- predict(rf\_model, newdata = test\_data)

*# Evaluate the model*

*# Calculate Mean Absolute Error (MAE)*

MAE\_rf <- mean(abs(predicted\_price\_rf - Price))

*# Calculate Root Mean Squared Error (RMSE)*

RMSE\_rf <- sqrt(mean((predicted\_price\_rf - Price)^2))

*# Calculate R-squared value*

R\_squared\_rf <- cor(predicted\_price\_rf, Price)^2

print(paste("Mean Absolute Error (MAE) with Random Forest:", MAE\_rf))

print(paste("Root Mean Squared Error (RMSE) with Random Forest:", RMSE\_rf))

print(paste("R-squared with Random Forest:", R\_squared\_rf))

saveRDS(rf\_model, file = "random\_forest\_model.rds")

# Shiny Training Code :

library(shiny)

library(shinydashboard)

library(DT)

library(randomForest) *# Assuming you trained your model using randomForest*

*# Load your trained model*

price = "price"

car\_data <- data.frame(

car\_name = c("Corolla", "Civic", "F-150", "Elantra", "Camaro"))

*# Define UI for application*

ui <- dashboardPage(

dashboardHeader(

title = div(

"Used Car Price Prediction",

tags$style(HTML("font-size: 24px;"))

)

),

dashboardSidebar(

*# Input fields*

width = "30%",

tags$head(

tags$style(

HTML(".sidebar .form-group.shiny-input-container {

width: 90%;

align-item:center;

margin-left:2rem;

}

.img

{

margin-bottom:3rem;

}

"

)

)

),

sidebarMenu(

menuItem("Home", tabName = "home", icon = icon("home")),

numericInput("year" ,"Year", min = 1900, max = 2019, value = 2015),

selectInput("brand", "Brand of Car",

choices <- c("Maruti", "Hyundai", "Honda", "Audi",

"Nissan", "Toyota", "Volkswagen", "Tata", "Land",

"Mitsubishi", "Renault", "Mercedes-Benz", "BMW",

"Mahindra", "Ford", "Porsche", "Datsun", "Jaguar",

"Volvo", "Chevrolet", "Skoda", "Mini", "Fiat", "Jeep",

"Smart", "Ambassador", "Isuzu", "Force", "Bentley",

"Lamborghini")),

selectInput("location", "Location of Car",

choices <- c("Mumbai", "Pune", "Chennai", "Coimbatore", "Hyderabad", "Jaipur", "Kochi", "Kolkata", "Delhi", "Bangalore", "Ahmedabad")),

numericInput("kilometer", "Kilometer Driven", value = 0),

selectInput("fuel", "Fuel Type", choices = c("Petrol", "Diesel", "CNG", "LPG")),

selectInput("transmission", "Transmission", choices = c("Manual", "Automatic")),

selectInput("owner", "Owner Type",

choices = c("First", "Second", "Third", "Fourth", "Test Drive Car")),

numericInput("mileage", "Mileage (kmpl)", value = 0),

numericInput("power", "Power (bhp)", value = 0),

numericInput("engine", "Engine (CC)", value = 0),

numericInput("seats", "Number of Seats", value = 0),

selectInput("car\_name", "Name of Car", choices = car\_data$car\_name)

)

),

dashboardBody(

*# Main panel for displaying results and the car image*

tabItems(

tabItem(tabName = "home",

fluidRow(

box(

title = "Prediction",

status = "primary",

solidHeader = TRUE,

width = 12,

height = "50%",

DTOutput("prediction")

),

box(

title = "Car Image",

status = "primary",

solidHeader = TRUE,

width = 12,

height = "50%",

div(

style = "display: flex; justify-content: center; align-items: flex-start;margin-bottom:6rem;margin-left:10rem;",

imageOutput("carImage")

)

),

box(

title = "Price Prediction",

status = "primary",

solidHeader = TRUE,

width = 12,

height = "50%",

textOutput("formatted\_price"),

)

)

)

)

)

)

*# Define server logic*

server <- function(input, output) {

car\_data <- data.frame(

car\_name = c("Corolla", "Civic", "F-150", "Elantra", "Camaro"),

image\_file = c("corolla.jpg", "civic.jpg", "f150.jpg", "elantra.jpg", "camaro.jpg")

)

*# Server logic for prediction*

output$prediction <- renderDT({

*# Creating a data frame with parameters and values*

data <- data.frame(

"Parameters" = c("Name of Car", "Year", "Brand", "Kilometer Driven", "Fuel Type", "Transmission", "Owner Type", "Mileage", "Power", "Engine", "Seats", "Price"),

"Values" = c(input$car\_name, input$year, input$brand, input$kilometer, input$fuel, input$transmission, input$owner, input$mileage, input$power, input$engine, input$seats, "A"),

stringsAsFactors = FALSE

)

*# Highlighting the row with "Price" equal to "A"*

data$Parameters <- ifelse(data$Parameters == "Price", price, data$Parameters)

*# Returning the data frame as a datatable*

datatable(data, rownames = FALSE, options = list(

columnDefs = list(

list(targets = "\_all", className = "valueColumn")

)

))

})

*# Dynamically render car image based on the selected car name*

output$carImage <- renderImage({

*# Get the selected car name*

selected\_car <- input$car\_name

*# Find the corresponding image file name based on the selected car name*

image\_file <- car\_data$image\_file[car\_data$car\_name == selected\_car]

*# If image file name is found, render the image*

if (!is.na(image\_file) && file.exists(paste0("www/", image\_file))) {

list(src = paste0("www/", image\_file), width = "70%" )

} else {

*# If image file is not found, display a placeholder image*

list(src = "www/placeholder.jpg", width = "65%")

}

}, deleteFile = FALSE)

*# Predict price using the trained model*

output$formatted\_price <- renderText({

*# Prepare input data for prediction*

*# Create new\_data dataframe with proper data types and levels*

new\_data <- data.frame(

Age = as.integer(2024 - input$year), *# Corrected the calculation of Age*

Location = factor(input$location, levels = levels(train\_data$Location)),

Kilometers\_Driven = as.integer(input$kilometer),

Fuel\_Type = factor(input$fuel, levels = levels(train\_data$Fuel\_Type)),

Transmission = factor(input$transmission, levels = levels(train\_data$Transmission)),

Owner\_Type = factor(input$owner, levels = levels(train\_data$Owner\_Type)),

Mileage = as.numeric(input$mileage),

Engine = as.integer(input$engine),

Power = as.numeric(input$power),

Seats = as.integer(input$seats),

Brand = factor(input$brand, levels = levels(train\_data$Brand))

)

cat("\n",names(new\_data),"\n")

cat(is.numeric(new\_data$Year))

cat(is.numeric(new\_data$Age))

cat(is.numeric(new\_data$Kilometers\_Driven))

cat(is.numeric(new\_data$Mileage))

cat(is.numeric(new\_data$Engine))

cat(is.numeric(new\_data$Power))

cat(is.numeric(new\_data$Seats))

cat(is.factor(new\_data$Location))

cat(is.factor(new\_data$Fuel\_Type))

cat(is.factor(new\_data$Transmission))

cat(is.factor(new\_data$Owner\_Type))

cat(is.factor(new\_data$Brand))

*# Make prediction using the loaded model*

predicted\_price <- predict(mymodel, new\_data)

predicted\_price = round(predicted\_price,2)

*# Format the predicted price with a range of +/- 2 Lacs*

formatted\_price <- paste(predicted\_price, "Lacs")

*# Return the formatted predicted price*

formatted\_price

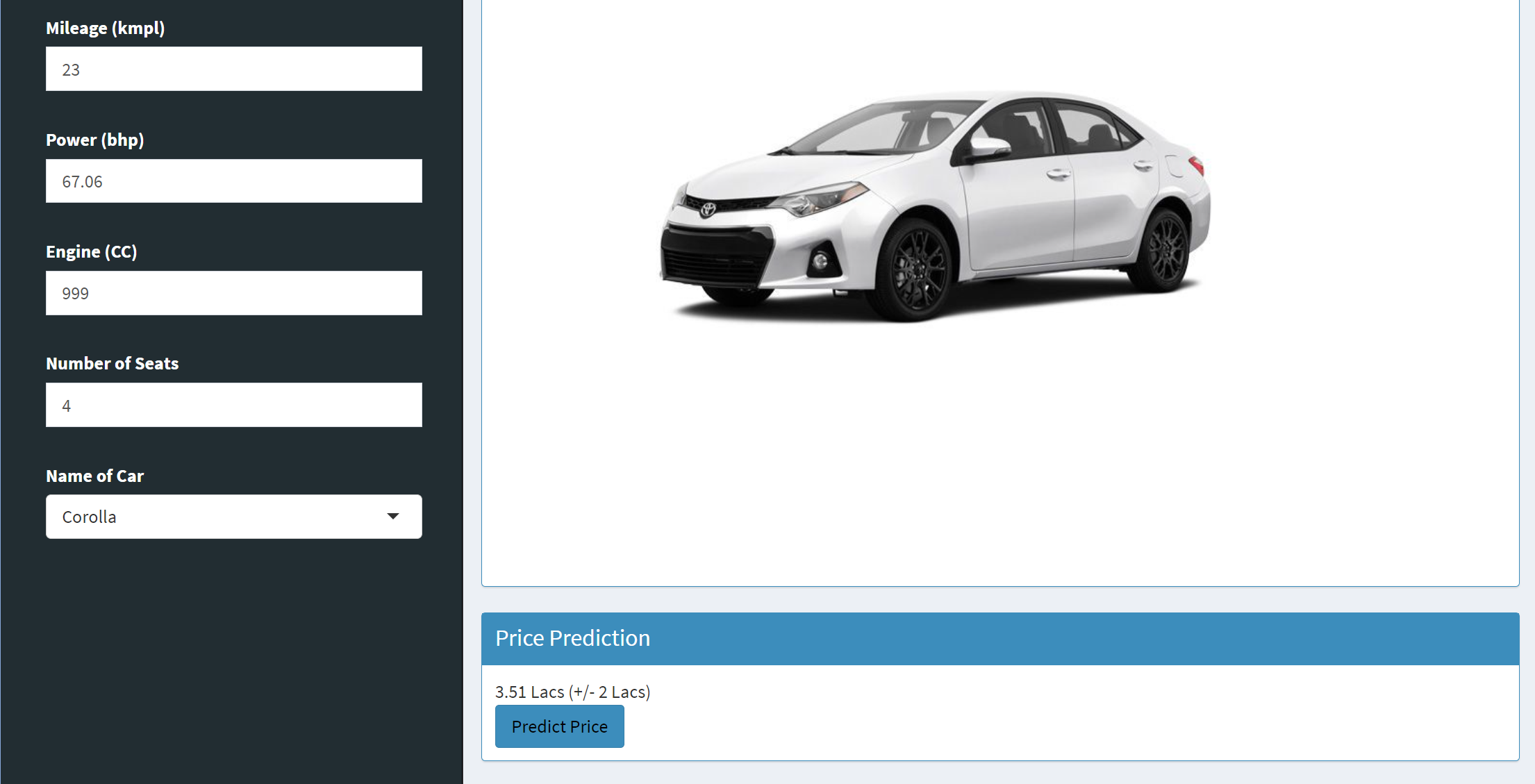
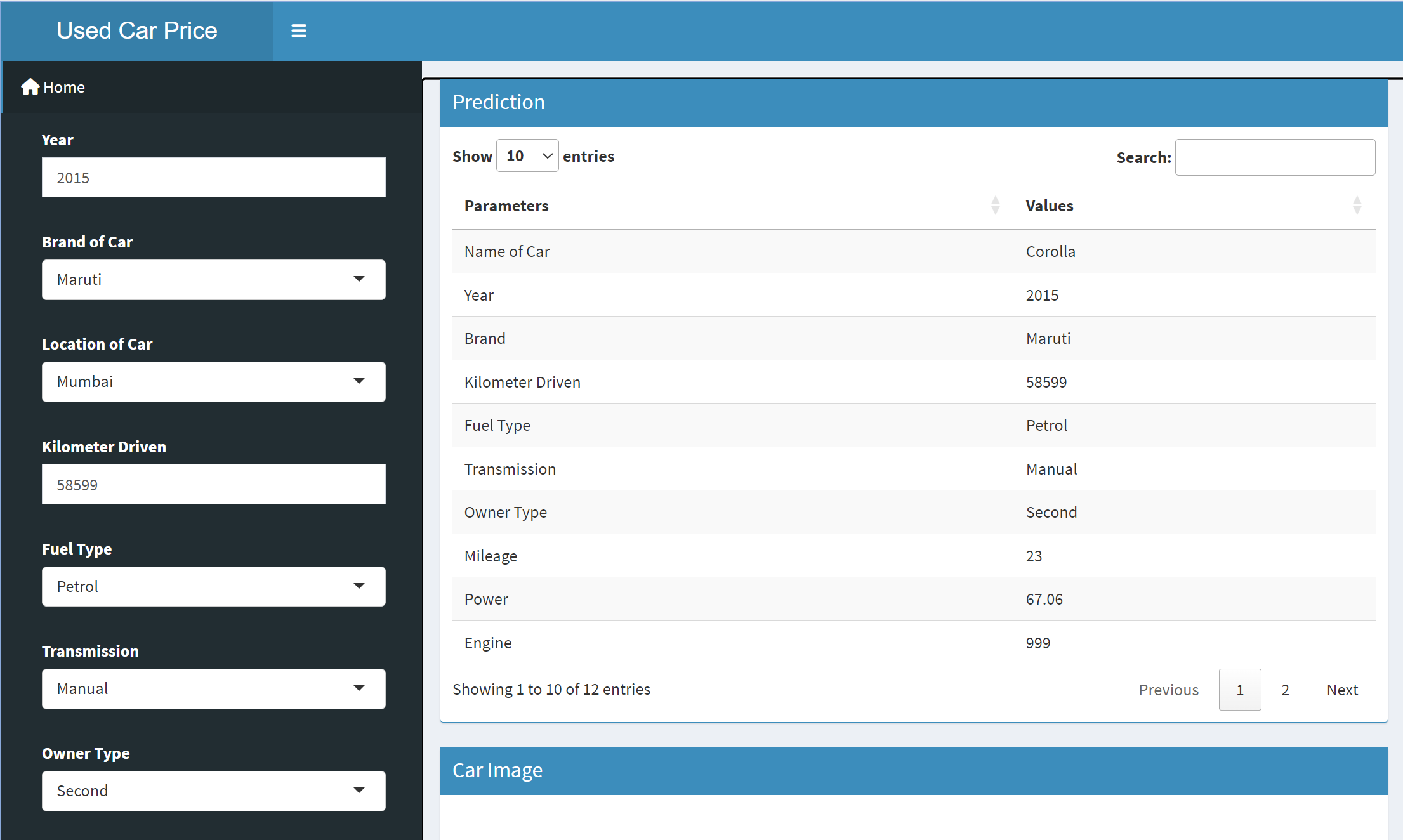
})

}

*# Run the application*

shinyApp(ui = ui, server = server)

1. Shiny App



# Fig ; Interface of Shiny app

# Evaluation Parameters : *Explain which evaluation parameters you have used in your project*.

# Results and Discussions:

Data Visualization :

1.

# WhatsApp Image 2024-04-10 at 17.26.27_8e68d773

# 2.

# WhatsApp Image 2024-04-10 at 17.26.28_7a2cb139

# 3.

# WhatsApp Image 2024-04-10 at 17.26.28_bc534344

# 4.

# WhatsApp Image 2024-04-10 at 17.26.28_4cc86287

# 5.

# kilo ownerkilo fuelkilo brandkilo trans

# 

# Fig : Feature Correlation Diagram

# 

# 15 .Conclusions:

# After conducting a comprehensive evaluation of various machine learning algorithms, it is evident that Random Forest emerges as the optimal choice for the predictive modeling task at hand. Random Forest surpasses Linear Regression, Decision Tree, and SVM (with radial) in accuracy. Its superiority is confirmed by lower MAE and RMSE, along with a higher R-squared value.

# The values of metrics are : Random Forest: MAE: 1.24104022 RMSE: 2.46771 R-Squared: 0.94924194 Based on this, the Random Forest algorithm has the lowest Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE), and it also achieves the highest R-squared value. Therefore, it appears to be the most accurate algorithm among the ones tested.

# References:

# [1] Kriswantara, B., & Sadikin, R. (2022). Used Car Price Prediction with Random Forest Regressor Model. *Journal of Information Systems, Informatics and Computing Issue Period*, *6*(1), 40–49.

# [2] MUTİ, S., & YILDIZ, K. (2023). Using Linear Regression For Used Car Price Prediction. *International Journal of Computational and Experimental Science and Engineering*, *9*(1), 11–16.

# [3] Gupta, V., M.L, S., & K.C, T. (2021). USED CAR PRICE PREDICTION. *International Journal of Multidisciplinary Advanced Scientific Research and Innovation*, *1*(10), 256–262.

# [4] -, D. A. G., -, P. S. S., -, Y. R. M., -, A. U. P., & -, S. G. T. (2023). Used Car Price Prediction Using Random Forest Algorithm. *International Journal For Multidisciplinary Research*, *5*(3).

# [5] Cui, B., Ye, Z., Zhao, H., Renqing, Z., Meng, L., & Yang, Y. (2022). Used Car Price Prediction Based on the Iterative Framework of XGBoost+LightGBM. *Electronics (Switzerland)*, *11*(18).

# [6] S, R., R, B. T., T, B. G., Hegde, R. P., & Ramesh, S. (2023). Used Car Price Prediction Using Machine Learning. *International Journal for Research in Applied Science and Engineering Technology*, *11*(5), 1176–1180.

# [7] Dahiya, H., Aggarwal, C., Goyal, S., & Agarwal, M. (2021). USED CAR PRICE PREDICTION USING MACHINE LEARNING. *International Journal of Multidisciplinary Advanced Scientific Research and Innovation*, *1*(10), 246–251.

# [8] Varshitha, J., Jahnavi, K., & Lakshmi, C. (2022). Prediction Of Used Car Prices Using Artificial Neural Networks And Machine Learning. In *2022 International Conference on Computer Communication and Informatics, ICCCI 2022*. Institute of Electrical and Electronics Engineers Inc.

# [9] Alhakamy, A., Alhowaity, A., Alatawi, A. A., & Alsaadi, H. (2023). Are Used Cars More Sustainable? Price Prediction Based on Linear Regression. *Sustainability*, *15*(2), 911.

[10] **Huang, J., Saw, S. N., Feng, W., Jiang, Y., Yang, R., Qin, Y., & Seng, L. S. (2023). A Latent Factor-Based Bayesian Neural Networks Model in Cloud Platform for Used Car Price Prediction. IEEE Transactions on Engineering Management.**[11] **PREDICTION PRICE OF USED CARS. (2023). International Research Journal of Modernization in Engineering Technology and Science. https://doi.org/10.56726/irjmets33331**

[12] **Shaprapawad, S., Borugadda, P., & Koshika, N. (2023). Car Price Prediction:An Application of Machine Learning. 6th International Conference on Inventive Computation Technologies, ICICT 2023 - Proceedings. https://doi.org/10.1109/ICICT57646.2023.10134142**

[13] **Zhu, Y. (2023). Prediction of the price of used cars based on machine learning algorithms. Applied and Computational Engineering, 6(1). https://doi.org/10.54254/2755-2721/6/20230917**

[14] **Venkatasubbu P. , Ganesh M. Used Cars Price Prediction using Supervised Learning Techniques . International Journal of Engineering and Advanced Technology (2019)**

[15] **1Amit Kewat , Nitesh Kanojiya . Price Prediction of Used Cars Using Linear Regression. Journal of Online Engineering Education**

[16] **Gegic, E., Isakovic, B., Keco, D., Masetic, Z., & Kevric, J. (2019). Car price prediction using machine learning techniques. TEM Journal, 8(1).** [**https://doi.org/10.18421/TEM81-16**](https://doi.org/10.18421/TEM81-16)

[17] **-, D. A. G., -, P. S. S., -, Y. R. M., -, A. U. P., & -, S. G. T. (2023). Used Car Price Prediction Using Random Forest Algorithm. International Journal For Multidisciplinary Research, 5(3).** [**https://doi.org/10.36948/ijfmr.2023.v05i03.3308**](https://doi.org/10.36948/ijfmr.2023.v05i03.3308)

[18] **Mrs Shyamali Das1 , Mr Ananta Laha2 , Mr Alok Jena3 , Ms Priyadarshini Samal4. Used Car Price Prediction Using Machine Learning Techniques. International Journal of Research Publication and Reviews**

[19] **Cui, B., Ye, Z., Zhao, H., Renqing, Z., Meng, L., & Yang, Y. (2022). Used Car Price Prediction Based on the Iterative Framework of XGBoost+LightGBM. Electronics (Switzerland), 11(18). https://doi.org/10.3390/electronics11182932**[20] **Monburinon, N., Chertchom, P., Kaewkiriya, T., Rungpheung, S., Buya, S., & Boonpou, P. (2018). Prediction of prices for used car by using regression models. Proceedings of 2018 5th International Conference on Business and Industrial Research: Smart Technology for Next Generation of Information, Engineering, Business and Social Science, ICBIR 2018. https://doi.org/10.1109/ICBIR.2018.8391177**