

# Senior Project

## Analysis of the Used Cars Market in Saudi Arabia Using Machine and Deep Learning

| Group Information   |         |
|---------------------|---------|
| Name                | ID      |
| Abdulrahman Simbawa | 1945140 |
| Marwan Al Ghamdi    | 1945856 |
| Saeed Qahas         | 1947614 |
| Abdullah Al Shareef | 2042708 |

Supervised By:

Dr. Omar Al-Ghushairy

## Table of Content

|   |           |
|---|-----------|
| <b>Abstract</b>   | <b>8</b>  |
| <b>Acknowledgement</b>                                      | <b>8</b>  |
| <b>1. Chapter 1: Introduction</b>                           | <b>9</b>  |
| 1.1 Introduction.....                                       | 9         |
| 1.2 Problem Definition .....                                | 9         |
| 1.3 The Recommended Solution.....                           | 9         |
| 1.4 Project Scope .....                                     | 10        |
| 1.4.1 Aims .....  | 10        |
| 1.4.1 Objectives.....                                       | 10        |
| 1.5 Target User .....                                       | 10        |
| 1.6 Methodology .....                                       | 11        |
| 1.7 Project Plan .....                                      | 13        |
| 1.8 The Gantt Chart .....                                   | 15        |
| 1.9 Tools and Requirements .....                            | 16        |
| 1.10 Conclusion .....                                       | 16        |
| 1.11 References.....  | 17        |
| <b>2. Chapter 2: Literature review</b>                      | <b>18</b> |
| 2.1 Context.....  | 18        |
| 2.2 Related Work .....                                      | 18        |
| 2.3 Comparison Between Proposed Systems and Literature..... | 21        |
| 2.4 Conclusion .....  | 22        |
| 2.5 References.....   | 23        |
| <b>3. Chapter 3: Data Collection</b>                        | <b>24</b> |
| 3.1 Introduction.....                                       | 24        |
| 3.2 Data Collection .....                                   | 24        |
| 3.3 Data Description .....                                  | 26        |
| 3.4 Exploratory Data Analysis .....                         | 27        |
| 3.5 Conclusion .....  | 32        |
| 3.6 References.....   | 32        |
| <b>4. Chapter 4: Data preparation &amp; preprocessing</b>   | <b>33</b> |
| 4.1 Introduction.....                                       | 33        |
| 4.2 Problems with the dataset .....                         | 33        |
| 4.3 Data preparation and Preprocessing.....                 | 35        |
| 4.4 Conclusion .....  | 38        |
| 4.5 References.....   | 38        |

|  |           |
|--|-----------|
| <b>5. Chapter 5: Model Building</b>                | <b>39</b> |
| 5.1 Introduction.....                              | 39        |
| 5.2 Experiments Setup and Tools .....              | 39        |
| 5.3 Initial Parameters and Selection Criteria..... | 40        |
| 5.4 Conclusion .....                               | 43        |
| 5.5 References.....                                | 43        |
| <b>6. Chapter 6: Results and Discussions</b>       | <b>44</b> |
| 6.1 Introduction.....                              | 44        |
| 6.2 Performance Evaluation Metrics.....            | 44        |
| 6.3 Experiments Results.....                       | 45        |
| 6.4 Discussion .....                               | 54        |
| 6.5 Conclusion .....                               | 54        |
| 6.6 References.....                                | 54        |
| <b>7. Chapter 7: Conclusion and Future Work</b>    | <b>55</b> |
| 7.1 Introduction.....                              | 55        |
| 7.2 Conclusion .....                               | 55        |
| 7.3 The Interactive Dashboard.....                 | 55        |
| 7.4 The Web Solution .....                         | 56        |
| 7.5 Difficulties and Limitations .....             | 58        |
| 7.6 Future Work.....                               | 58        |

## List of Figures

|  |    |
|--|----|
| Figure 1.6.1, A view of a car offered on syarah.com, along with details associated with it ..  | 11 |
| Figure 1.6.2, A brief review of the methodology taken with the project .....   | 12 |
| Figure 1.8.1, The Gantt Chart for the project .....  | 15 |
| Figure 3.2.1, Example of the data structure of Syarah.com website .....  | 24 |
| Figure 3.4.1, A box plot figure that shows the year attribute distrubtion.....   | 27 |
| Figure 3.4.2, A box plot figure that shows the engine size attribute distribution .....  | 28 |
| Figure 3.4.3, A box plot figure that shows the mileage attribute distribution (we can notice that this attribute needs some cleaning)..... | 28 |
| Figure 3.4.4, A box plot figure that shows the price attribute distribution.....   | 29 |
| Figure 3.4.5, A visualization shows the correlation between numeric attributes .....   | 29 |
| Figure 3.4.6, A figure showing the regions with their count .....  | 30 |
| Figure 3.4.7, This figure shows the count of each brand in the data .....  | 30 |
| Figure 3.4.8, This figure contains multiple bar charts graphs that show the top 5 brands in the data with their car models .....           | 31 |
| Figure 3.4.9, A figure shows the frequencies of different categorical attributes in the data ...   | 32 |
| Figure 4.2.1, Shows the unique colors .....  | 33 |
| Figure 4.2.2, A bar chart showing the count of the car brands.....   | 34 |
| Figure 4.2.3, A figure shows the unique regions name in the data .....   | 34 |
| Figure 4.2.4, Toyota brand models.....   | 35 |
| Figure 4.3.1, A bar chart showing the new count of brands .....  | 35 |
| Figure 4.3.2, Shows the count of each color after combining .....  | 36 |
| Figure 4.3.3, A figure showing the new maximum and minimum numbers of the Mileage attribute .....  | 36 |
| Figure 4.3.4, A figure showing the new minimum and maximum numbers of the Price attribute .....  | 37 |
| Figure 4.3.5, A figure showing the new unique regions names.....   | 37 |
| Figure 4.3.6, Toyota brand models after cleaning.....  | 37 |
| Figure 5.2.1, A screenshot from Spyder IDE Shows the libraries imported.....   | 39 |
| Figure 5.3.1., The three tree model parameters .....   | 40 |
| Figure 5.3.2, The two Gradient Boosting Regressors parameters.....   | 41 |
| Figure 5.3.3, Cat Boost Regressor parameters .....   | 41 |
| Figure 5.3.4, K Nearest Neighbors parameters.....  | 41 |
| Figure 5.3.5, Support Vector Regression and Linear Regression parameters.....  | 42 |
| Figure 5.3.6, Deep Learning model parameters.....  | 42 |
| Figure 6.3.1, Visualization of the R2 accuracy for the <b>Decision Tree Regressor</b> model .....  | 46 |
| Figure 6.3.2, Visualization of the R2 accuracy for the <b>Extreme Gradient Boosting Regressor</b> model .....                              | 46 |
| Figure 6.3.3, Visualization of the R2 accuracy for the <b>Gradient Boosting Regressor</b> model .....                                      | 47 |
| Figure 6.3.4, Visualization of the R2 accuracy for the <b>Random Forest Regressor</b> model ...  | 47 |
| Figure 6.3.5, Visualization of the R2 accuracy for the <b>Linear Regression</b> model .....  | 48 |
| Figure 6.3.6, Visualization of the R2 accuracy for the <b>Extra Trees Regressor</b> model .....  | 48 |
| Figure 6.3.7, Visualization of the R2 accuracy for the <b>Cat Boost Rgressor</b> model.....  | 48 |

|  |    |
|--|----|
| Figure 6.3.8, Visualization of the R2 accuracy for the <b>Support Vector Regressor</b> model ...         | 49 |
| Figure 6.3.9, Visualization of the R2 accuracy for the <b>K Nearest Neighbours Regressor</b> model ..... | 49 |
| Figure 6.3.10, Visualization of the R2 accuracy for the <b>Deep Learning</b> model.....                  | 50 |
| Figure 7.3.1, The Interactive Dashboard .....  | 55 |
| Figure 7.4.1, The First Page of The Web-solution .....   | 56 |
| Figure 7.4.2, The Prediction Page .....  | 57 |
| Figure 7.4.3, A Sample Output for a Prediction .....   | 57 |

## List of Tables

|  |    |
|--|----|
| Table 3.31-, A table that shows the data attributes with the description. ....   | 26 |
| Table 3.4-1, this table concludes the basic statistical analysis of the numeric attributes in the dataset.....             | 27 |
| Table 6.3-1, A table shows the detailed results of the 10 models on the data. ....   | 45 |
| Table 6.3-2, A table shows the detailed results of the 10 models on the Toyota cars data .....                             | 50 |
| Table 6.3-3, A table shows the detailed results of the 10 models on the Nissan cars data.....                              | 51 |
| Table 6.3-4, Table 6.3 3, A table shows the detailed results of the 10 models on the GMC cars data .....                   | 51 |
| Table 6.3-5, A table shows the detailed results of the 10 models on the Mercedes cars data .                               | 51 |
| Table 6.3-6, A table shows the detailed results of the 10 models on the Kia cars data.....                                 | 52 |
| Table 6.3-7, A table shows the detailed results of the 10 models on the Chevrolet cars data.                               | 52 |
| Table 6.3-8, A table shows the detailed results of the 10 models on the Hyundai cars data...                               | 52 |
| Table 6.3-9, A table shows the detailed results of the 10 models on the Ford cars data .....                               | 53 |
| Table 6.3-10, A table shows the detailed results of the 10 models on the Lexus cars data.....                              | 53 |
| Table 6.3-11, A table shows the detailed average results of the 10 models on all the 9 separated car brand sets data ..... | 53 |

## List of Equations

|   |    |
|---|----|
| Equation 4.3-1, Standard Scaler Equation .....                | 37 |
| Equation 6.2-1, R Squared Equation.....                       | 44 |
| Equation 6.2-2, Mean Absolute Error Equation .....            | 44 |
| Equation 6.2-3, Mean Absolute Percentage Error Equation ..... | 45 |

## Abstract

Many people and businesses in the car sales industry in Saudi Arabia set the prices of used cars based either on intuition, asking experts in car sales, or other ways. The problem with these solutions is that they may lead to estimations that vary drastically from the prices that the cars are worth. To eliminate the price-setting problem, we can use historic data about the used cars that are being sold through websites that sell cars in Saudi Arabia to analyze the market and develop a solution that can help stakeholders predict accurate prices for used cars based on several inputs that affect the price. A machine learning model can predict continuous values, such as the price in our case, via several regression algorithms. Also, using deep learning techniques may give better accuracies than these models. As for the market study, an interactive dashboard will help summarize the most affecting factors on car prices. In this paper, we examine the use and application of multiple supervised machine learning and deep learning algorithms to predict how much a car is worth in Saudi Arabia. The predictions are based on data collected using web scrapping techniques from several sites that offer the service of buying and selling used cars in Saudi Arabia, such as Syarah.com, and Yallamotor.com. Examined algorithms varied among Decision tree, Xgboost, Gradient Boosting, Random Forest, Extra Trees, etc. The algorithms are then evaluated and compared using R squared score, Mean Absolute Error, and Mean Absolute Percentage Error to choose the best-performing algorithm out of the ten. All the algorithms scored a very high accuracy score. The Extra Trees algorithm was the best-performing algorithm with an R squared score of %95.7, a Mean Absolute Error of 10188.7, and a Mean Absolute Percentage Error of %14.7. consequently, this model will be deployed to a web application that can serve as a used car price estimator.

## Acknowledgement

We would like to express our special thanks and gratitude to Dr. Omar Al-Ghushairy our supervisor, for his guidance and advice given to us during the preparation of this paper. We also thank the committee responsible for evaluating graduation projects for their advice when we first presented our project to them.



# 1. Chapter 1: Introduction

## 1.1 Introduction

Owning a car these days has become very important for almost everyone, you need to have your car to transport from one place to another and fulfill your obligations. Some people would like to buy a brand-new car from the dealership and some other people would like to save money and buy a used car. The company that originally produces the cars set the prices of their cars based on several factors, from car manufacturing costs to shipping costs, etc. However, how can you set an accurate price for a car after it has been used for a while by someone? How can we know what factors affect car prices the most?

## 1.2 Problem Definition

Many people and businesses in the car sales industry in Saudi Arabia set the prices of used cars based either on intuition, asking experts in car sales, or other ways. The problem with these solutions is that they may lead to estimations that vary drastically from the prices that the cars are worth. Additionally, there are not many recourses that study the used cars market in Saudi Arabia, therefore, the factors that affect the prices of cars are mainly unclear.

## 1.3 The Recommended Solution

To eliminate the price-setting problem, we can use historic data about the used cars that are being sold through websites that sell cars in Saudi Arabia to analyze the market and develop a solution that can help stakeholders predict accurate prices for used cars based on several inputs that affect the price. A machine learning model can predict continuous values, such as the price in our case, via several regression algorithms. Also, using deep learning techniques may enhance the accuracy of these models. As for the market study problem, an interactive dashboard that summarizes the most affecting factors on car prices.

## 1.4 Project Scope

### 1.4.1 Aims

Our work in this project will be aimed to achieve the following:

- 1- Help stakeholders in the used car industry set prices in the best way possible.
- 2- Help decision-makers and stakeholders in understanding the used cars market in Saudi Arabia.

### 1.4.1 Objectives

The objectives that we aspire to fulfill to satisfy our aims are as follows:

- 1- Acquiring data on used cars across websites in Saudi Arabia.
- 2- Identifying the factors that affect the price of a car the most.
- 3- Building several regression machine learning models to predict the prices of cars and choosing the most accurate model.
- 4- Applying Deep Learning techniques to enhance the model built if possible.
- 5- Analyzing each car brand with the regression models individually and comparing them to make a solid and reliable study of the machine learning model used to predict car prices in Saudi Arabia.
- 6- Building a web solution to deploy our model on.
- 7- Building an interactive dashboard to derive insights into the used cars market in Saudi Arabia.

## 1.5 Target User

Our project target users and stakeholders are as follows:

- 1- Individuals that are willing to buy or sell used cars.
- 2- Businesses that buy or sell used cars.
- 3- Entrepreneurs that are willing to join the car industry.
- 4- Decision-makers studying the car market and industry in Saudi Arabia.

## 1.6 Methodology

Data was collected from websites that provide the service of selling and buying used cars in Saudi Arabia. The websites are [Syarah.com](https://syarah.com), [carswitch.com](https://carswitch.com), and [Yallamotor.com](https://yallamotor.com). We used the same methodology on all three websites and to explain the methodology, we will use the [Syarah.com](https://syarah.com) website as an example. [Syarah.com](https://syarah.com) is a website that enables individuals and companies to sell and buy new and used cars in Saudi Arabia. An individual can offer up his/her car on the website by registering a user and putting up their car details and specifications along with the price associated with it.

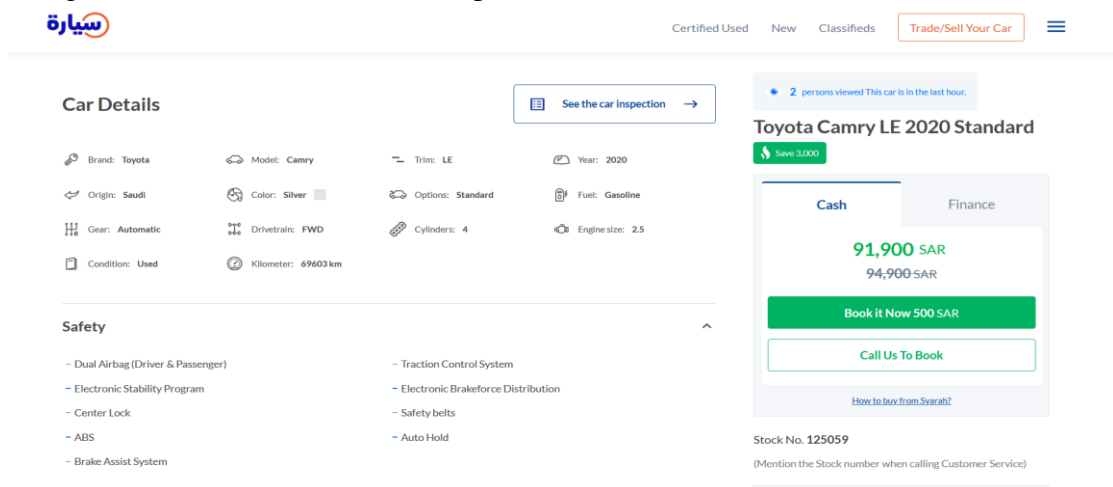


Figure 1.6.1, A view of a car offered on syarah.com, along with details associated with it

As shown in Figure 1.6.1, each car has a lot of details in the Cars Details box, which is mainly what affects the price of a car whether it is used or new. On our part, scrapping those details and prices into raw data, for us to analyze them, was the first step in the project and data collection process. The cars offered on the website were about 11000 cars, after removing the duplicates, the count was reduced to 8478 cars. To achieve our objectives, the three main platforms used are “Python”, “R”, and “Tableau”. Firstly, “Python” is an open-source, high-level programming language, that has simple syntax and data structures compared to other programming languages. Python, also, is enriched by many libraries that help developers achieve many purposes. In our case, Python can help us by using libraries, such as:

- 1- Pandas: is an open-source library that is built on top of python, the purpose is to manipulate data.
- 2- Numpy: a package in python that is the main purpose is to perform statistical and mathematical operations on data.
- 3- Matplotlib: is a python library used for making visualizations.
- 4- Scikit-learn: is an open-source python library whose main goal is for predictive analysis and applying machine learning algorithms.

5- TensorFlow: is a platform that helps in applying machine learning and artificial intelligence algorithms.

Secondly, there is “R”, which is a programming language that is very powerful in applying statistical operations and is effective in data analytics.

Thirdly, “Tableau”, is a software that enables its users to ensure high-quality data analysis and visualize results. Furthermore, it is a very powerful tool for building dashboards of the highest quality.

Finally, "Heroku" is a container-based cloud Platform as a Service (PaaS). Developers use Heroku to deploy, manage, and scale modern apps.

To identify all the inputs, processing, and outputs of our projects with the tools used to achieve all the objectives, it would be as follows:

- **Inputs:** car data, details, and prices scrapped and collected using a scrapper in R studio and collected from [Syarah.com](http://Syarah.com), [carswitch.com](http://carswitch.com), and [Yallamotor.com](http://Yallamotor.com).
- **Processing:** Cleaning the data and preprocessing it, using R studio and python. Also, applying machine learning algorithms to predict prices using Python along with its libraries.
- **Output:** A prediction of the price of a car based on the inputs on a web solution uploaded to the cloud using Heroku service to be open for use to the public. Furthermore, a dashboard that shows details about the car market in Saudi Arabia.

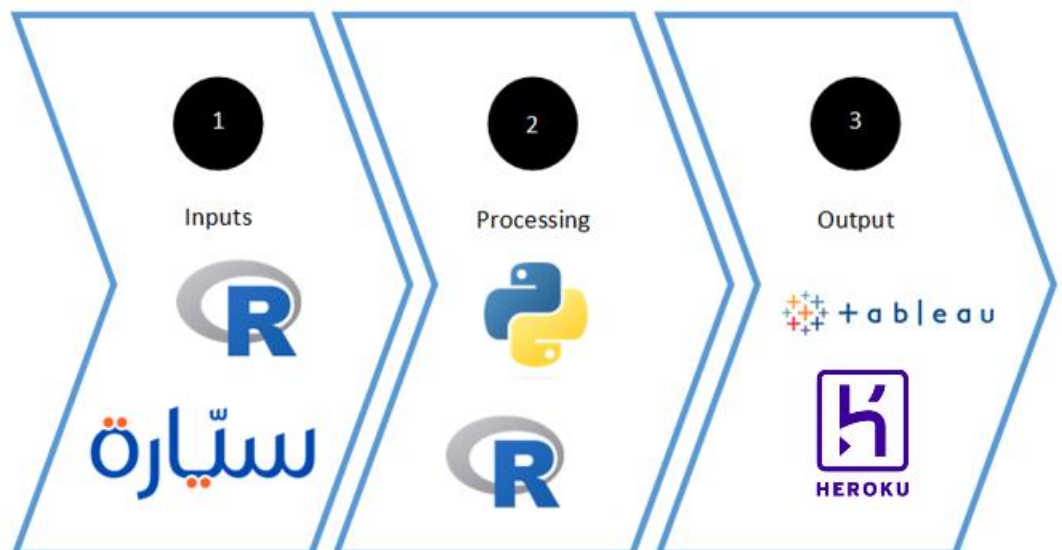


Figure 1.6.2, A brief review of the methodology taken with the project

## 1.7 Project Plan

Our Project Plan Consists of 6 phases, and we are willing to finish all these phases in six months divided into two semesters. We will finish the first three phases in the first semester, and the other three phases we will finish them in the second semester:

In the first phase, we are going to define the project in detail.

### **Phase 1: Defining the Project (From 11/9/2022 To 6/10/2022): -**

- 1 – Defining the Problem & the Recommended Solution.
- 2 – Defining the Project Scope.
- 3 – Defining the Project Methodology.
- 4 – Defining any Similar Scientific Papers to our Project.

In the second phase, we will start collecting the data from online websites, describe the data in detail, and perform exploratory data analysis on the data.

### **Phase 2: Data Collection (From 9/10/2022 To 20/10/2022): -**

- 1 – Collecting the data using a web scrubbing tool.
- 2 – Describing the Data.
- 3 – Exploratory Data Analysis.

Then, in phase 3, We will start preprocessing the data.

### **Phase 3: Data Preprocessing (From 23/10/2022 To 3/11/2022): -**

- 1 – Removing any untreated attributes.
- 2 – Removing any outliers.
- 3 – Fill in missing values.
- 4 – Feature Engineering.

After that, in phase 4, we will start building our model, interactive dashboard, and web application.

**Phase 4: Building the Model, Creating the Dashboard, and Building the Web Application (From 4/12/2022 To 20/1/2023): -**

- 1 – Building the Machine Learning and Deep Learning Models.
- 2 – Testing the Models and Choosing the best.
- 3 – Creating an Interactive Dashboard.
- 4 – Creating a Web Application.
- 5 – Deploy the Final Model and the dashboard to the Web Application.
- 6 – Testing the Web Application.

Finally, in phase 5, we schedule a day to discuss our project results with the Evaluation Committee, and then after reviewing the project and fixing any errors, the project will be ready to be published

**Phase 5: Results, Discussions, and Publishing (From 22/1/2023 To 2/2/2023): -**

- 1 – Discuss the Project Results with the Evaluation Committee.
- 2 – Publishing the Project Paper and the Web Application.

## 1.8 The Gantt Chart

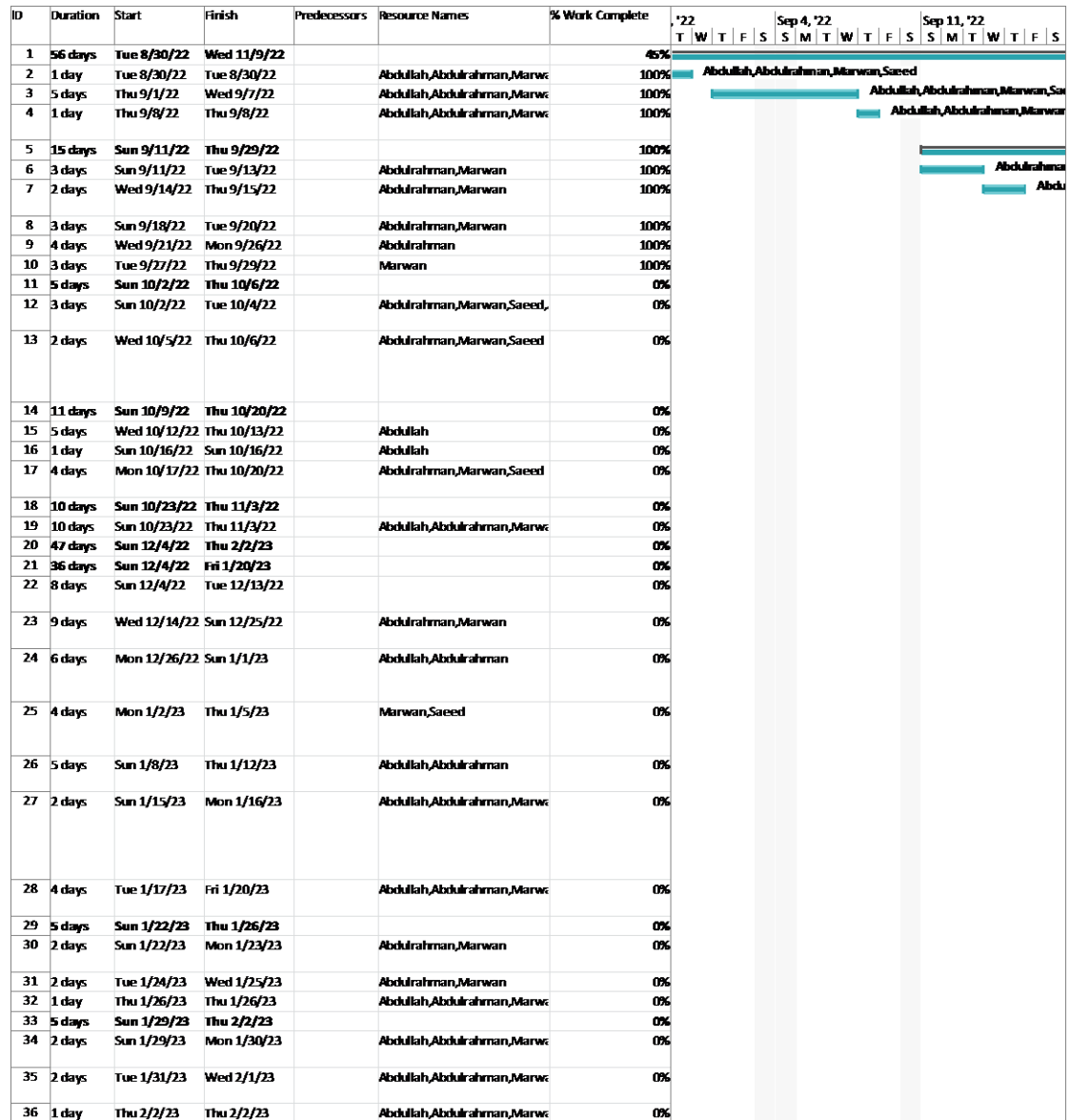


Figure 1.8.1, The Gantt Chart for the project

## 1.9 Tools and Requirements

The tools that we are going to use to fulfill our requirements while working on the project are:

- 1- Python programming language
- 2- R
- 3- RapidMiner
- 4- Tableau or Power BI
- 5- JavaScript & HTML
- 6- Heroku

Requirements:

- 1- We need to gather data from multiple websites that provide the service of selling and buying used cars.
- 2- Clean and analyze the data.
- 3- Building machine and deep learning models.
- 4- Create an interactive dashboard to show insight into the used cars market in Saudi Arabia.
- 5- Deploy our model to a web application.

## 1.10 Conclusion

In conclusion, the project focuses on solving problems of ambiguity in the used car industry in Saudi Arabia and predicting a price for a car by analyzing data scrapped from websites that sell used cars, such as [Syarah.com](http://Syarah.com). Then, make a dashboard that helps in deriving insights and makes it easier for decision-makers and entrepreneurs to have an idea of Saudi Arabia's car market at a glance. Furthermore, developing a web-based solution that utilizes the data collected about used cars to predict the price of a car based on several factors taken as input that goes into a model with a high accuracy rate. This project is aimed to target businesses, entrepreneurs, and individuals interested in Saudi Arabia's used car market.



## 1.11 References

[Syarah.com](http://Syarah.com)

[carswitch.com](http://carswitch.com)

[Yallamotor.com](http://Yallamotor.com)

[Business Intelligence and Analytics Software \(tableau.com\)](http://Business Intelligence and Analytics Software (tableau.com))

[Cloud Application Platform | Heroku](http://Cloud Application Platform | Heroku)

[R: The R Project for Statistical Computing \(r-project.org\)](http://R: The R Project for Statistical Computing (r-project.org))

[Welcome to Python.org](http://Welcome to Python.org)

[scikit-learn: machine learning in Python — scikit-learn 1.2.1 documentation](http://scikit-learn: machine learning in Python — scikit-learn 1.2.1 documentation)

[pandas - Python Data Analysis Library \(pydata.org\)](http://pandas - Python Data Analysis Library (pydata.org))

[NumPy](http://NumPy)

[TensorFlow](http://TensorFlow)

[Matplotlib documentation — Matplotlib 3.6.3 documentation](http://Matplotlib documentation — Matplotlib 3.6.3 documentation)

## 2. Chapter 2: Literature review

### 2.1 Context

The car market in Saudi Arabia is growing faster and faster. This market is confined to online sites that allow individuals to offer their cars or buy cars from other individuals. Another way of car transactions is offering or buying cars from dealerships. One example of a website that carries out car transactions is [Syarah.com](http://Syarah.com), which is a site that allows individuals and businesses to buy and sell cars.

Unfortunately, the market in Saudi Arabia is seemingly unclear in terms of used car prices and the factors that define these prices. Furthermore, the studies conducted on the used car market in Saudi Arabia are fairly limited or sparse. Due to the aforementioned reasons, an individual or a business wanting to make a car transaction would face difficulties estimating these prices and would face a lot of unclarity in determining how the used car market in Saudi Arabia is. An individual or a business might under/overestimate the price of the car. Hence, arises the importance of a study on the used car market in Saudi Arabia.

The prediction of a used car price might seem simple in the beginning. However, requires a lot of steps to be taken to find the factors that affect a car's price the most. Consequently, making a machine learning model that considers these factors without compromising the accuracy of the model would impose a challenge.

The solution proposed in this paper is to make a supervised machine learning model that utilizes regression algorithms, such as Support Vector Regressor (SVR), to make high-accuracy predictions of car prices. Moreover, the use of deep learning techniques might also help in making the accuracy of these predictions higher. Additionally, the final model would be deployed into a web solution. Not to mention, building a dashboard that can help any decision-maker understand the car market in Saudi Arabia.

### 2.2 Related Work

The number of papers that predict the price of used cars or secondhand cars is decent and each paper is impressive in its way. In his paper, Sameerchand (2014) used 4 machine-learning models to predict the prices of used cars in Mauritius. He used a linear regression model, K-Nearest Neighbors model, Decision Tree model, and Native Bayes model. The best one was the linear regression model which had high regression coefficient with Nisan and Toyota cars mostly. The main limitation of his study was the low number of records that were used.[1]

In another paper, Mukkesh & Pattabiraman (2019) used supervised learning techniques to predict the prices of used cars based on past consumer data. In their idea, they used three machine-learning models: Lasso Regression, Multiple Regression, and Regression Tree. The results between the three models weren't significantly different

from each other, but the models with the highest accuracy were the Lasso Regression and the Multiple Regression models. To get higher accuracies, they suggested using more advanced machine learning models like the Random Forest Model.[2]

Saamiyah & Nushrah & Sameerchand (2015), used Artificial Neural Networks and machine learning to predict the price of secondhand cars. They collected the data from various websites and newspapers and used 4 approaches: Linear regression, K-Nearest Neighbors, Support Vector Regression, and Neural Networks. The best among these approaches was the Support Vector Regression followed by a multilayer perceptron with back-propagation (Neural Networks) and the worst one was the K-Nearest Neighbors.[3]

Sait & Arzu (2020), tried to predict the prices of secondhand cars using Artificial Neural Networks and collected the data from a website on the internet. They calculated the correlation, mean absolute error, and mean absolute percentage error values to make a prediction. After that, they performed a data mining process which included: Problem Statement, Data Preparation, Exploration, Modeling, Evaluation & Deployment, and got a pretty good result.[4]

Ozer & Omer (2019), explained why a machine learning technology could be second-hand to estimate secondhand car prices. They found the dataset on an auction website on the internet and after cleaning the data they started building their model. After determining the values that significantly affect the price, they deployed a linear regression model and got a high R square accuracy aka 89.1%.[5]

Fahad & Akash & Ahnaf & Sifat (2021), examined the use of machine learning techniques to predict the prices of pre-owned cars in Bangladesh. They collected the data from famous a website in Bangladesh that offer many services and one of them is selling and buying cars using the web scrubbing tool. After collecting the data, they preprocessed it, explored it, and then started building the model. They tried 5 different machine learning models: Linear Regression, Lasso Regression, Decision Tree, Random Forest, and Extreme Gradient Boosting, and the best one was Extreme Gradient Boosting. In the end, they deployed their model to a web application in a local machine so it can be later made available to end users.[6]

Murat (2016), investigated the performance of Artificial Neural Networks in predicting the prices of secondhand cars. He collected the data from an item-selling website in a CSV format. Then, he started preprocessing the data which included the normalization of raw data, and then started feature exploring and selection. In the Neural Network process, he created the model using 9 features and 720 instances. The model has one hidden layer with 15 neurons and used the Levenberg-Marquardt backpropagation Function. After running the model, the calculated mean absolute percentage error is 8.28%, and he expressed that the main limitation of his study was that there is only one type of car model in his collected data.[7]

Enci & Anni & Haoran & Tao (2022), tried to predict the prices of used cars in the view of the PSO-GRA-BPNN neural network model and compared it with other models from other papers. They collected the data from a famously used car trading platform in China. After that, they started preprocessing the data and used grey relational analysis to filter the feature variables of factors affecting used car prices. After building their model, they compared it with the traditional BPNN model and the multiple linear regression, random forest, and support vector machine regression models proposed by other researchers. Their model was better with an MAE equal to 0.475, R accuracy equal to 0.998, and R Square accuracy equal to 0.984. The main limitation of their model is that it takes too much time to get these high-accuracy results.[8]

K.Samruddhi & Dr. R.Ashok (2020), proposed a supervised machine learning model to predict the prices of used cars. They found the dataset on Kaggle, and then they preprocessed and explored it, and started building the model. They tried the K-Nearest Neighbors and the linear regression models. To find the optimal number of K's for the K-Nearest Neighbors model, they tried different values of K's from 2 to 10 and measured the distance using the Euclidean Distance metric. The best model was the K-Nearest Neighbors with an optimal value of k equal to 4, with a Root Squared Error of 4.01, Absolute Mean Error of 2.01, and an accuracy of 85%. The linear regression model didn't perform well, the accuracy was 71%.[9]

Muhammad Asghar, Khalid Mehmood, Samina Yasin, and Zimal MehboobKhan (2021) used data collected from Kaggle to predict the prices of used cars using a machine learning algorithm which was Linear Regression. They used a Statistical test to get the design value of P and get the optimal features. After preprocessing the data and testing the model, the model had an R2 accuracy equal to 90% which is very good.[10]

Mehmet Bilen (2021) aimed to determine the best prediction model for second-hand car prices in turkey by using heuristic algorithms. The algorithms he used were Linear Regression, SVM, and ANN algorithms. The dataset in his study was collected from 4 different sites that provide second-hand sales advertisements over the internet. He created a new dataset including car features and price information by compiling the advertisements on the sites that provide car buying and selling advertisement services over the internet, and he shared this dataset for researchers to use in the model development phase. As a result of the prediction processes using different preprocessing steps and different prediction algorithms, the Fisher+ANN model achieved the best performance with MAE 0.01050, MSE 0.000281 error, and an R2 accuracy of 89% performance value.[11]

Nandini Mazumdar (2021) tried to predict the prices of used cars in online marketplaces. She collected the data from Kaggle consisting of 370,000 cases, scraped from e-Bay Kleinanzeigen. After preprocessing the data, she analyzed it using multiple linear regression with standard predictor entry and got pretty good results. The power

level for here study was >99% and adequate and the recommended N:p ratio of 15:1 was met. She conducted that the limitation of her study lies in the specific characteristics of vehicles used in Germany, as e-Bay classified listings were local to that region.[12]

Baoyang Cui, ZhonglinYe, Haixing Zhao, Zhuome Renqing, Lei Meng, and Yanlin Yang (2022) tried to predict the prices of used cars in China using an iterative framework combining XGBoost and LightGBM. The data used in the research are derived from the car valuation training data and verification data used in the 2021 MathorCup big data competition, with a total of 30,000 training data. They applied the framework to different models some of them were: Random Forest, Linear Regression, XGB, and LGBM. The results of the models without the framework were low, but after adding the XGB+LGBM iterative framework, the results were much better. It was verified that the XGB+LGBM iteration framework can be applied to other models and greatly improve the performance of the original model.[13]

Kshitij Kumbar, Pranav Gadre, and Varun Nayak (2021) tried to predict the prices of used cars by developing machine learning models that can accurately predict the price of a used car based on its features. They collected data about used car sales from all over the United States, and the data is available on Kaggle. After preprocessing the data and applying the models, The results show that the Random Forest model and K-Means clustering with linear regression yield the best results, but are compute-heavy, However, Random Forests tend to overfit the dataset due to the tendency of growing longer trees. For better performance, they plan to judiciously design deep learning network structures, use adaptive learning rates, and train on clusters of data rather than the whole dataset, to correct the overfitting in Random Forest.[14]

Anu Yadav, Ela Kumar, and Piyush Kumar Yadav (2021) examined the use of machine learning models to accurately predict the price of a second-hand car according to its parameter or characteristics. The data they used for the research was from Kaggle, the data is based on used vehicles, especially cars. After exploring and preprocessing the data, they tested the models. The outcome of their research shows that clustering with linear regression and the Random Forest model yield the best accuracy outcome.[15]

## 2.3 Comparison Between Proposed Systems and Literature

All the papers that we included above are impressive and different from each other, and all of them have different objectives, but they all share one similar objective which is to predict used car prices. Each paper used a different process for collecting the data and preprocessing the data and the same is applied to the machine learning and deep learning models used. In the aspect of data collection, some of the papers used web scrapping to collect the data they needed from famous websites in their country like in Fahad & Akash & Ahnaf & Sifat paper [6] or, Enci & Anni & Haoran & Tao

paper [8]. Other papers used data that had already been collected like Muhammed & Khalid & Samina & Zimal paper [10]. None of these papers collected data about the used car market in Saudi Arabia. In this project, we will use web scrapping to collect data from a famous trading website in Saudi Arabia called syarah.com.

Then, in the aspect of building the model, some papers tried only machine learning models like Sameerchand paper [1], K.Samruddhi & Dr. R.Ashok paper [9], and some papers tried only deep learning models like Murat paper [7], and then there only one paper that tried both machine learning and deep learning models which is Saamiyah & Nushrah & Sameerchand paper [3]. The Saamiyah & Nushrah & Sameerchand paper [3] tried only five different machine learning models and compared them with a deep learning model. In this proposed project, we will use different machine learning models like Linear Regression, Multiple Linear Regression, Random Forest, Support Vector Regression, Lasso Regression, K-Nearest Neighbor, Decision Tree, and more. After that, we will build a deep learning model and compare it with different machine learning models and enhance the model built.

Furthermore, we will deploy our model to a web application so it can be used by end users in Saudi Arabia. In Fahad & Akash & Ahnaf & Sifat paper [6], they also deployed their model to a web application on a local machine. In this proposed project, we will add more variables that need to be filled in by the end users on the web application, so it can predict the price of the car more and more accurately.

Additionally, we will build an interactive dashboard that helps understand and derive insights from the used cars market in Saudi Arabia.

## 2.4 Conclusion

To conclude, after reviewing literature related to our work, it was found that the first difference between our system and the literature was the data collection phase, which included methods like a web scrapper that scrapped the cars' data from [Syarah.com](http://Syarah.com), [carswitch.com](http://carswitch.com), and [Yallamotor.com](http://Yallamotor.com). Also, this imposes the importance of the market's main location for [Syarah.com](http://Syarah.com) and the other websites, which is Saudi Arabia. Nearly, all the related research was based on markets outside Saudi Arabia. Another difference in our system from the literature is the features that are used as independent variables to train the machine learning model and the deep learning model. Furthermore, the proposed system uses deep learning techniques, which are not abundant in a lot of literature. Additionally, the system proposed includes a dashboard that shows multiple visualizations that analyzes the car business in Saudi Arabia, which enables stakeholders and decision-makers to draw useful insights, that help growing businesses in the car industry.

## 2.5 References

- [1] [\(PDF\) Predicting the Price of Used Cars using Machine Learning Techniques \(researchgate.net\)](#)
- [2] [\(PDF\) Used Cars Price Prediction using Supervised Learning Techniques \(researchgate.net\)](#)
- [3] [Predicting the Price of Second-hand Cars using Artificial Neural Networks | Sameerchand Pudaruth - Academia.edu](#)
- [4] [Journal of Internet Applications and Management » Submission » Price estimation of secondhand cars sold on the internet with artificial neural network method \(dergipark.org.tr\)](#)
- [5] [Avrupa Bilim ve Teknoloji Dergisi » Submission » Prediction of The Prices of Second-Hand Cars \(dergipark.org.tr\)](#)
- [6] [Information | Free Full-Text | Application of Machine Learning Techniques to Predict the Price of Pre-Owned Cars in Bangladesh \(mdpi.com\)](#)
- [7] [\(PDF\) Secondhand Car Price Estimation Using Artificial Neural Network \(researchgate.net\)](#)
- [8] [Sustainability | Free Full-Text | Research on the Prediction Model of the Used Car Price in View of the PSO-GRA-BP Neural Network \(mdpi.com\)](#)
- [9] [Used Car Price Prediction using K-Nearest Neighbor Based Model \(researchgate.net\)](#)
- [10] [Used Cars Price Prediction using Machine Learning with Optimal Features | Pakistan Journal of Engineering and Technology \(theskyjournal.net\)](#)
- [11] [\(PDF\) Predicting Used Car Prices with Heuristic Algorithms and Creating a New Dataset \(researchgate.net\)](#)
- [12] [Predicting Used Car Prices in Online Marketplaces Linear Regression Approach \(researchgate.net\)](#)
- [13] [Electronics | Free Full-Text | Used Car Price Prediction Based on the Iterative Framework of XGBoost+LightGBM \(mdpi.com\)](#)
- [14] [\[PDF\] CS 229 Project Report: Predicting Used Car Prices | Semantic Scholar](#)
- [15] [Object detection and used car price predicting analysis system \(UCPAS\) using machine learning technique | Linguistics and Culture Review \(lingcure.org\)](#)



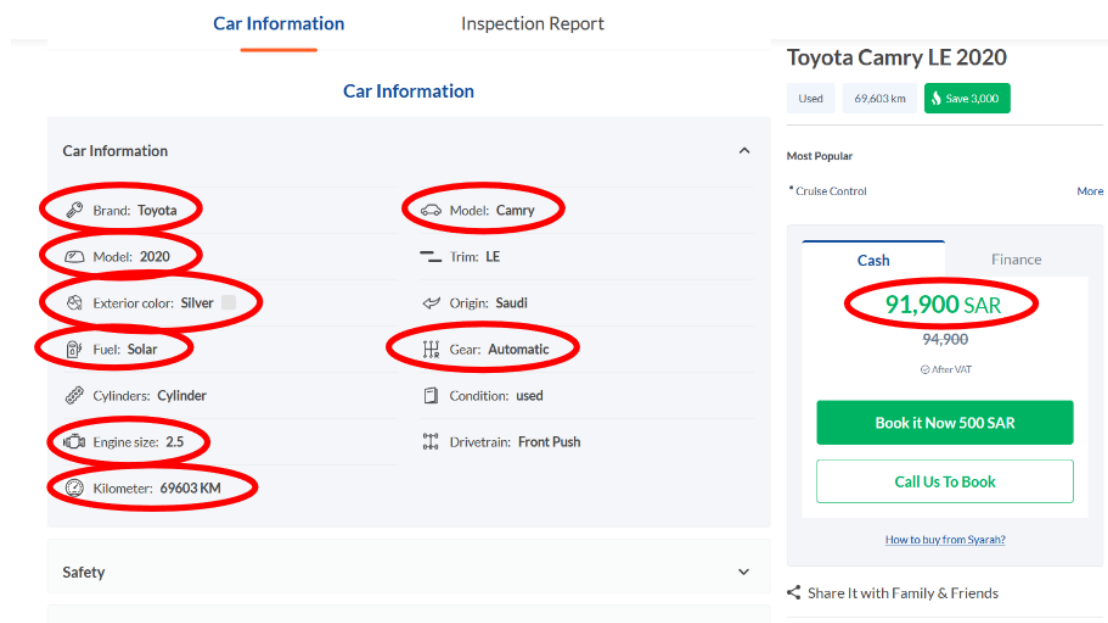
## 3. Chapter 3: Data Collection

### 3.1 Introduction

Data collection is the most critical phase in any Data Science project. It is where you start collecting data for your project and it is essential because if you didn't collect trustworthy and dated data, your project results would be unreliable. In this chapter, we will discuss how we collected our data, what tools we used, from where we collected the data, and what difficulties we had. Then, we will list every attribute from the data and describe each one of them. Next, we will perform an Exploratory Data Analytics process and share some findings and visualizations.

### 3.2 Data Collection

Firstly, the data collected for the project was from famous websites in Saudi Arabia which are [Syarah.com](https://syarah.com), [carswitch.com](https://carswitch.com), and [Yallamotor.com](https://yallamotor.com). these websites enable individuals and dealerships in Saudi Arabia to sell or buy new or used cars. We selected these websites to collect the data from them because of some important factors. One of these factors is data trustworthiness, which means these websites contain reliable, trusted, and dated data. Another factor is the data structure, those websites contain very structured and organized data. One more factor is data completeness, which means the values on the three websites like car color for instance are filled by



The screenshot displays the 'Car Information' section of the Syarah.com website for a Toyota Camry LE 2020. The 'Car Information' section is highlighted with red circles around the following attributes:

- Brand: Toyota
- Model: Camry
- Model: 2020
- Exterior color: Silver
- Fuel: Solar
- Gear: Automatic
- Engine size: 2.5
- Kilometer: 69603 KM

The right sidebar shows the car's price at 91,900 SAR and a 'Book it Now 500 SAR' button.

Figure 3.2.1, Example of the data structure of Syarah.com website

the users.

Secondly, the data from these websites were collected using web-scraping tools. The tools used were packages and libraries in R language that enable us to web-scrap the data. We used three libraries in R which is:



### 1 – dplyr

A library that is used for data manipulation, cleaning, summarizing, and providing a consistent set of verbs that helps in solving many data manipulation challenges [1].

### 2 - rvest

An R package that allows us to scrape data and information from an HTML web page and read it into R [2].

### 3 - reticulate

An R package that allows R to talk to Python and work within RStudio. It also provides functionality to manage multiple python installations [3].

Also, while we were web-scraping the data we used some methods like CSS Selector, Tag Name, and XPath to identify the attributes we want to collect.

Thirdly, we had some difficulties while collecting the data. One of these difficulties was that the web pages were constantly changing which leads to the appearance of multiple patterns that must be dealt with. Another difficulty was the appearance of anomalous values because of the free writing of the specification on the websites. Generally, because of these difficulties, the web-scraping process took a long time to work.

### 3.3 Data Description

In this section, the data collected will be described through a data dictionary, which will help us understand the meaning of each variable or column in the dataset used.

| Column/Variable Name | Column/Variable Description   | Example   |
|----------------------|---|-----------|
| Brand                | This column contains the manufacturing car company  | Toyota    |
| Model                | This column contains the name of the model that the manufacturing company set for the car | Accent    |
| Year                 | This column contains the year in which the car has been manufactured in                   | 2018      |
| Color                | This column contains the description of the car's exterior color                          | Gray      |
| Fuel_Type            | This column contains the type of fuel that the car runs on                                | Gas       |
| Gear_Type            | This column contains the type of gear that is implemented in the car.                     | Automatic |
| Engine Size          | This column contains the size of the car engine   | 3.5       |
| Mileage              | This column contains the distance that the car has crossed in Kilometers                  | 86500     |
| Region               | This column contains the city from which the car is offered from                          | Riyadh    |
| Price                | This column contains the price that the car bears in Saudi Riyals                         | 95000     |

*Table 3.3 1-, A table that shows the data attributes with the description.*

### 3.4 Exploratory Data Analysis

In this phase, an overall exploration and visualizations would be provided on each attribute, along with a table that shows some statistical methods that were applied to explore data. The main tools used to explore the data are the R and Python languages. These languages were chosen mainly because it is powerful statistically and can provide visualizations easily.

Initially, the rows in the data are 8103 rows and 10 columns.

| Attribute   | Minimum | 1 <sup>st</sup> Quartile | Median | Mean   | 3 <sup>rd</sup> Quartile | Maximum  | Standard Deviation | Number of Null values |
|-------------|---------|--------------------------|--------|--------|--------------------------|----------|--------------------|-----------------------|
| Year        | 1964    | 2015                     | 2017   | 2016   | 2019                     | 2023     | 4.54               | 0                     |
| Engine_Size | 1       | 1.6                      | 2.7    | 3.01   | 4                        | 9        | 1.42               | 421                   |
| Mileage     | 1       | 55000                    | 100000 | 128750 | 175126                   | 20000000 | 294794.6           | 0                     |
| Price       | 5000    | 40500                    | 65000  | 94988  | 120000                   | 1300000  | 90444.01           | 308                   |

Table 3.4-1, this table concludes the basic statistical analysis of the numeric attributes in the dataset

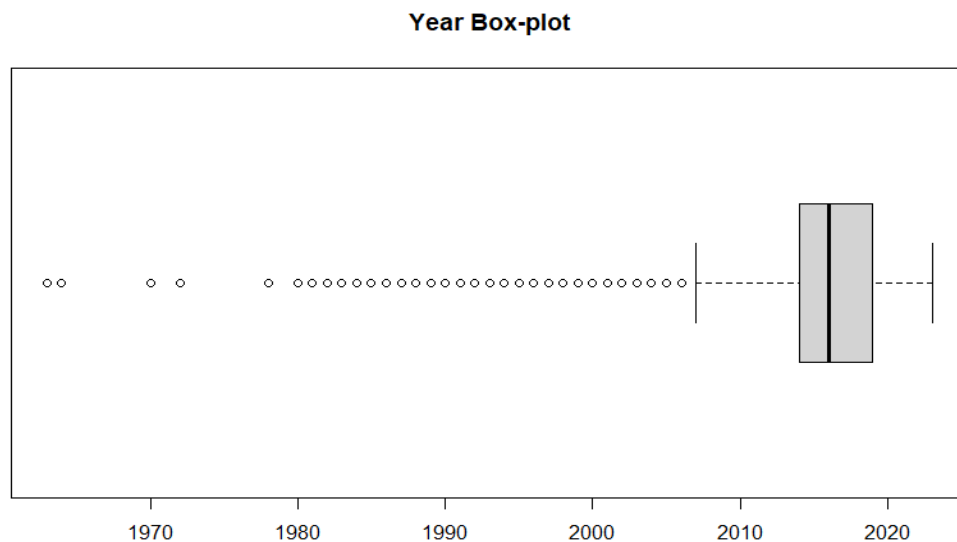


Figure 3.4.1, A box plot figure that shows the year attribute distribution

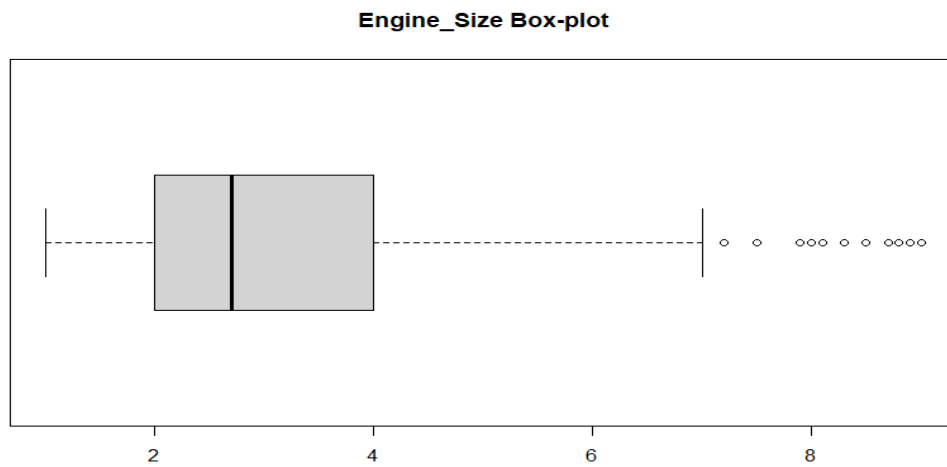


Figure 3.4.2, A box plot figure that shows the engine size attribute distribution

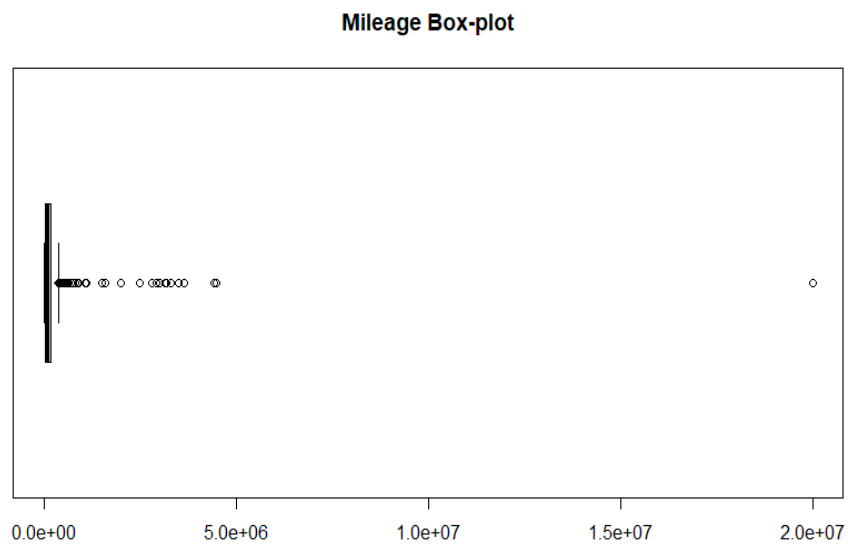


Figure 3.4.3, A box plot figure that shows the mileage attribute distribution (we can notice that this attribute needs some cleaning)

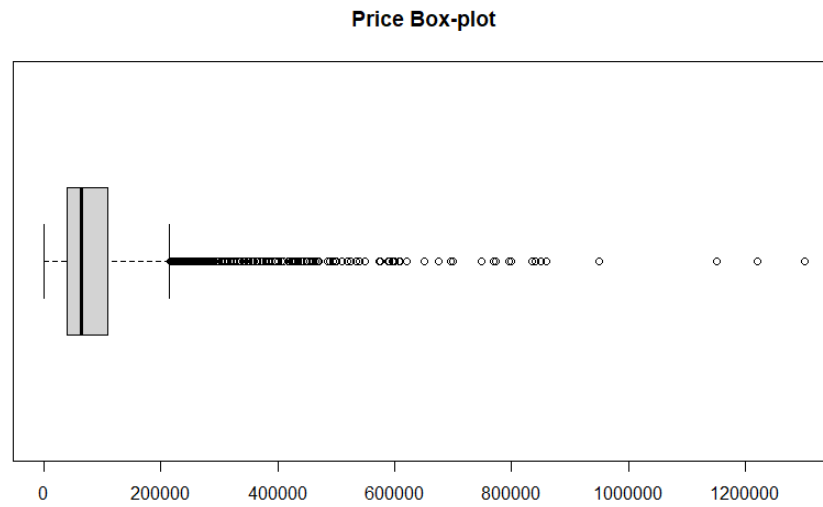


Figure 3.4.4, A box plot figure that shows the price attribute distribution

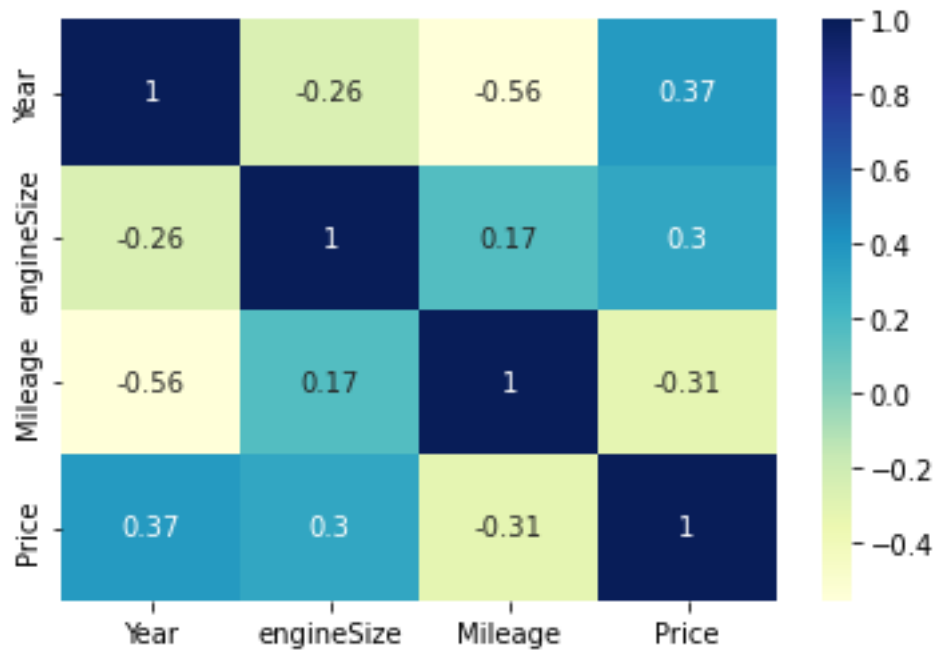


Figure 3.4.5, A visualization shows the correlation between numeric attributes

## Regions

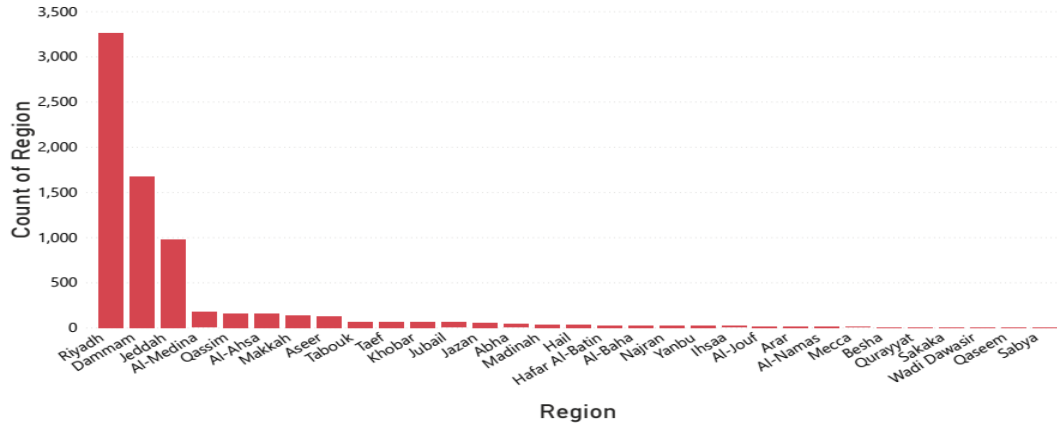


Figure 3.4.6, A figure showing the regions with their count

## Brands

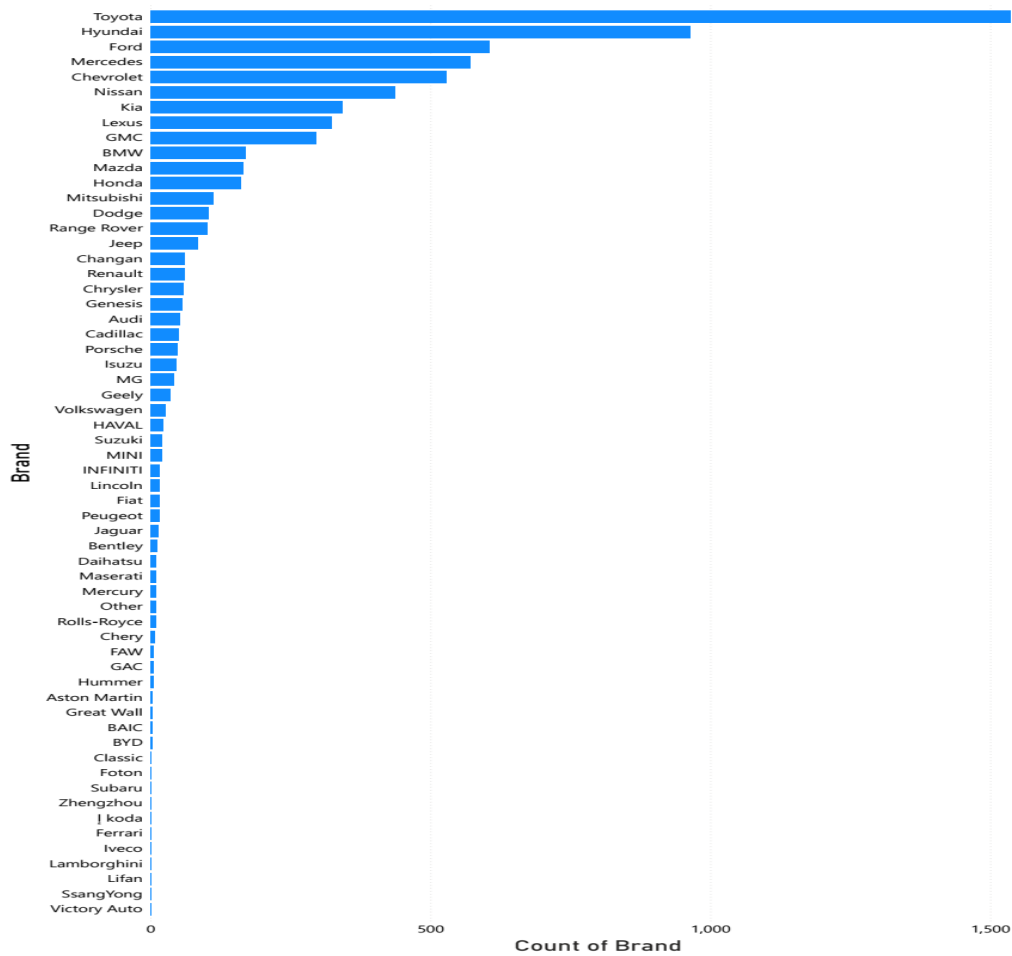


Figure 3.4.7, This figure shows the count of each brand in the data

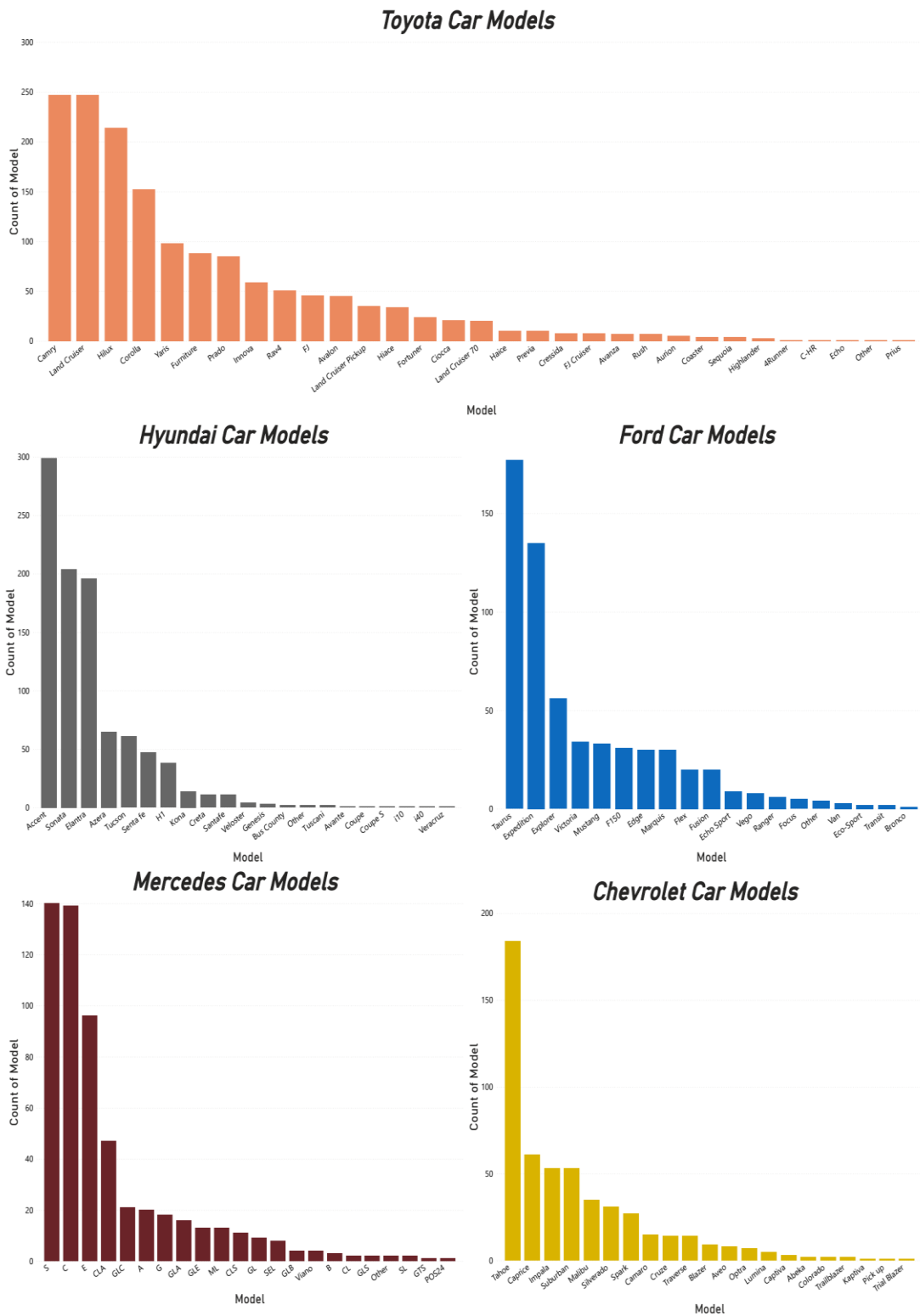


Figure 3.4.8, This figure contains multiple bar charts graphs that show the top 5 brands in the data with their car models

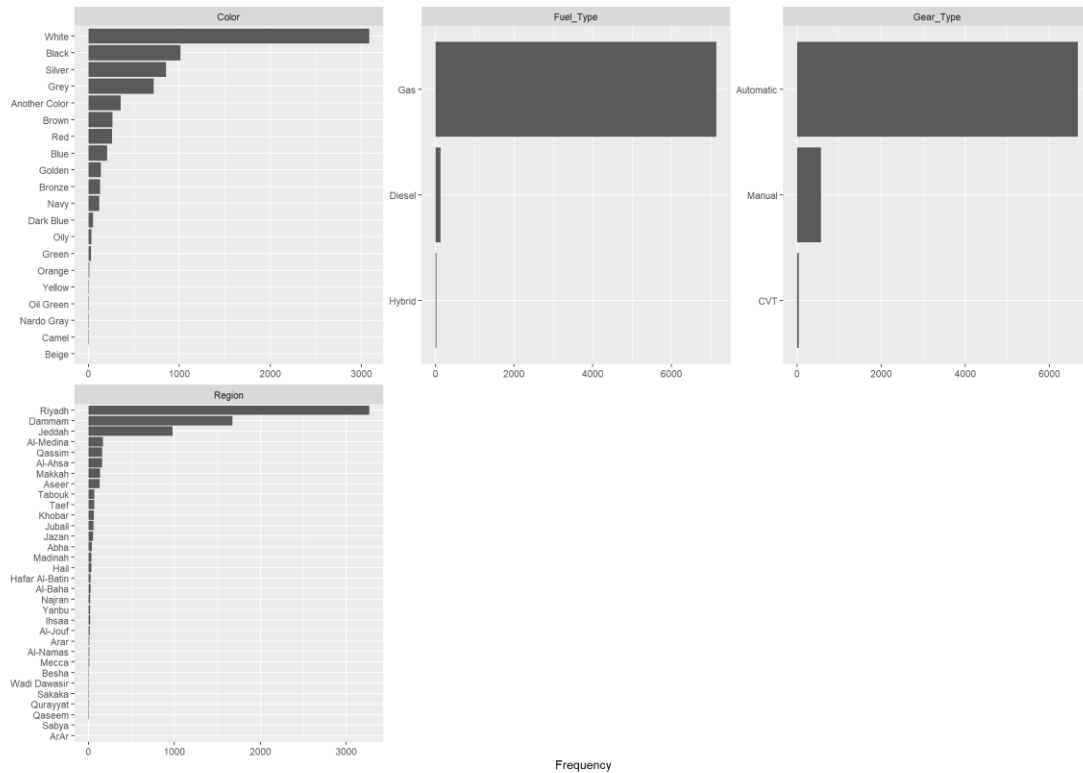


Figure 3.4.9, A figure shows the frequencies of different categorical attributes in the data

### 3.5 Conclusion

In conclusion, this chapter discusses the sources which the data was collected from, such as [Syarah.com](http://Syarah.com), which is a website that enables individuals and dealerships to offer and buy used or new cars. Consequently, the techniques used to scrap the data from the websites included web-scraping with the tools in R such as `dplyr`[1], `rvest`[2], and `reticulate`[3]. Moving on, the attributes that were chosen to be included in the dataset, which consist of 8103 rows and 10 attributes, were identified and described thoroughly in the data dictionary. Lastly, the performed exploratory data analysis techniques showed plots, the correlation between variables, and other statistical methods that are used to understand the data further.

### 3.6 References

- [1] [A Grammar of Data Manipulation • dplyr \(tidyverse.org\)](https://dplyr.tidyverse.org/)
- [2] [rvest: easy web scraping with R - Posit](https://rvest.tidyverse.org/)
- [3] [Getting started with Python using R and reticulate | R-bloggers](https://www.r-bloggers.com/getting-started-with-python-using-r-and-reticulate/)



## 4. Chapter 4: Data preparation & preprocessing

### 4.1 Introduction

After acquiring the data, there are a lot of challenges and problems that need to be fixed in the dataset. In this chapter, these problems will be discussed, as well as the solution to solve these problems, along with the tools used to solve these problems. Additionally, the dataset has some attributes that need to be tuned in a way that suits the models that are going to be used. The attributes that are set to be tuned will be discussed, along with the tools and methods that are going to be used to tune them.

### 4.2 Problems with the dataset

In this section, the problems with the dataset will be discussed by each attribute, meaning that each column problem will be provided. The main tools used in identifying the problems with the dataset and solving them will be R language and Python. The library used is mainly Pandas, which is a library in Python programming language that enables data manipulation easily [1].

**Color:** The problem with the color column is that some colors have a low count, which might hurt the accuracy of the models. Also, some of these colors like “Camel” and “Beige” as shown in figure 4.2.1, are nearly identical to the eye, hence there is no point in making them separate. Another problem is the "Another color" values as shown in figure 4.2.1, which is not having a specific color value, which means it may be any other color value.

|             |                 |              |          |             |
|-------------|-----------------|--------------|----------|-------------|
| "Black"     | "Silver"        | "Grey"       | "Navy"   | "White"     |
| "Bronze"    | "Another Color" | "Golden"     | "Brown"  | "Blue"      |
| "Red"       | "Oily"          | "Green"      | "Orange" | "Yellow"    |
| "Dark Blue" | "Camel"         | "Nardo Gray" | "Beige"  | "Oil Green" |

*Figure 4.2.1, Shows the unique colors*

**Engine\_Size:** The only problem with the engine size is that there are 421 null values in this column.

**Mileage:** The only problem with mileage is that there are a lot of very clear outliers. These outliers can be detected easily by looking at [table 3.4.1](#) and figure 3.4.3, which observe that there is a max value of 20000000 and a very high standard deviation (294794.6).

**Brand:** The problem with the brand attribute as shown in figure 4.2.2, is that many brands are very low in their count, and we plan to analyze each brand individually. So, analyzing those brands with a low count is not going to give us a reliable result.

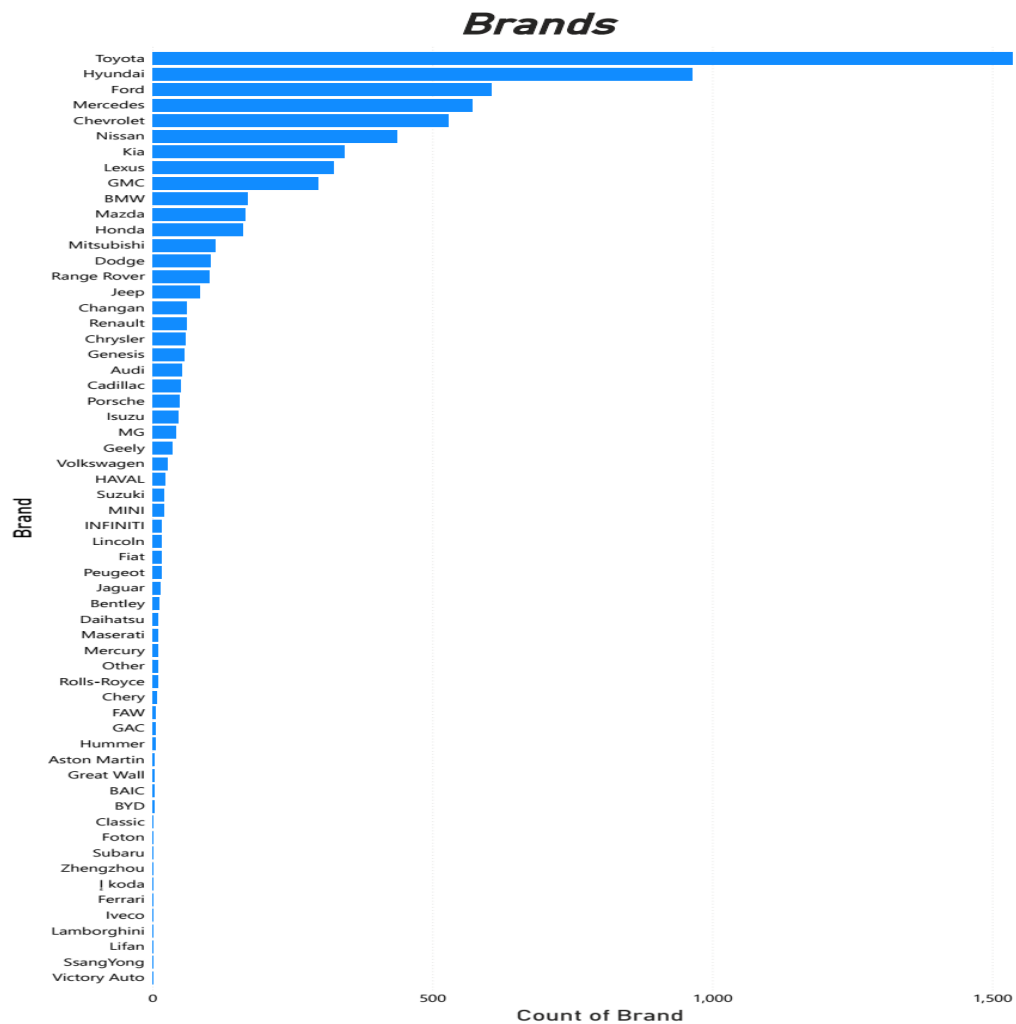


Figure 4.2.2, A bar chart showing the count of the car brands

**Price:** The price also has a problem with the outliers, this can be shown by looking at [table 3.4.1](#), which observes that the standard deviation (90444.01) is very high. Also, there are 307 null values in the “Price” attribute that need to be dealt with.

**Region:** We have too many regions and city names in the data, some of them refer to the same region or city but with different names.

|                  |          |            |                |           |
|------------------|----------|------------|----------------|-----------|
| "Riyadh"         | "Jeddah" | "Dammam"   | "Al-Medina"    | "Qassim"  |
| "Makkah"         | "Jazan"  | "Tabouk"   | "Aseer"        | "Al-Ahsa" |
| "Taef"           | "Sabya"  | "Khobar"   | "Abha"         | "Al-Baha" |
| "Yanbu"          | "Hail"   | "Al-Namas" | "Jubail"       | "Al-Jouf" |
| "Hafar Al-Batin" | "Najran" | "Arar"     | "Wadi Dawasir" | "Besha"   |
| "Qurayyat"       | "Sakaka" | "Ihsaa"    | "Mecca"        | "Qaseem"  |
| "Madinah"        | "ArAr"   |            |                |           |

Figure 4.2.3, A figure shows the unique regions name in the data

**Model:** the problem with the model attribute is that we have too many models for each car brand refers to the same model but with different names.

|                       |              |            |                   |
|-----------------------|--------------|------------|-------------------|
| "Land Cruiser"        | "Yaris"      | "Camry"    | "Corolla"         |
| "Prado"               | "Furniture"  | "Aurion"   | "Rav4"            |
| "Hilux"               | "FJ"         | "Avalon"   | "Ciocca"          |
| "Land Cruiser Pickup" | "Cressida"   | "Innova"   | "Land Cruiser 70" |
| "Previa"              | "Rush"       | "Echo"     | "Avanza"          |
| "Hiace"               | "C-HR"       | "Coaster"  | "Prius"           |
| "4Runner"             | "FJ Cruiser" | "Fortuner" | "Haice"           |
| "RAV4"                | "other"      | "Sequoia"  | "Highlander"      |
| "LAND CRUISER 70"     |              |            |                   |

Figure 4.2.4, Toyota brand models

As shown in figure 4.2.4, we can see that we have car models in the Toyota brand with the same names, like the LAND CRUISER 70.

### 4.3 Data preparation and Preprocessing

In this section, we will tune, organize, and clean the data by solving the problems of the dataset that we addressed earlier in the previous section. By performing this process properly, the data will be tuned and ready for deployment on it. The better the data processing, the more accurate and reliable the project results will be.

First, we will start with the **Brand** attribute. In the **Brand** attribute, we removed every brand with a count of less than 200. Keeping these brands with low counts is not going to give us a solid result, as we planning to analyze each brand individually. So, we removed them and kept only the brands with high counts which we are going to focus on them.

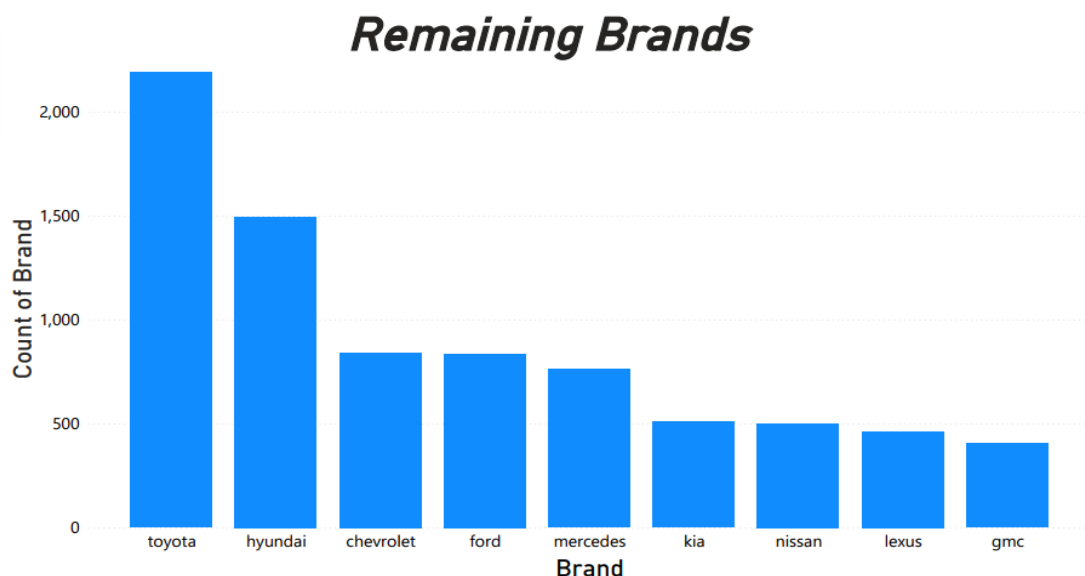


Figure 4.3.1, A bar chart showing the new count of brands

For the **Color** attribute, we combined those colors with low counts with other superior similar colors. We saw that combining those colors with the low count is a better option than removing them. An example of this is the color "Navy" and color "Blue" both of them are similar to each other, so we combined them.

The list of colors we combined are:

- "Navy" and "Dark Blue" are combined with their superior color "Blue".
- "Beige" and "Camel" are combined with their superior color "Brown".
- "Golden" is combined with its superior color "Yellow".
- "Oil Green" and "Oily" are combined with their superior color "Green".
- "Nardo Gray" is combined with its superior color "Gray".

Then we have the "Another Color" issue, so we decided to replace all color values that are addressed as "Another Color" with the most occurred color value which is "White".

|       |      |       |      |       |     |       |        |
|-------|------|-------|------|-------|-----|-------|--------|
| black | blue | brown | gray | green | red | white | yellow |
| 1004  | 342  | 408   | 1993 | 59    | 248 | 3780  | 168    |

*Figure 4.3.2, Shows the count of each color after combining*

In the **Engine\_Size** attribute, we imputed all 421 null values by filling them with the median value of the engine size of each car model in each brand. Meaning that if a record with a null engine size value and the model for instance is Camry and the brand is Toyota, the null value is filled with the median value of the engine size in the Camry model. Models have different median values and each car with a null engine size value is filled with the median value of its model.

For the **Mileage** attribute, we fixed the outliers problem by removing any record with a **Mileage** value of more than 900000 km and less than 100 km. Also, we removed any record within a year range of 2013 and less with a **Mileage** value less than 10000 km, because it is not reasonable to have the car from 2013 and less with this low value.

|      |         |        |        |         |        |
|------|---------|--------|--------|---------|--------|
| Min. | 1st Qu. | Median | Mean   | 3rd Qu. | Max.   |
| 101  | 56425   | 101000 | 128374 | 176608  | 880000 |

*Figure 4.3.3, A figure showing the new maximum and minimum numbers of the Mileage attribute*

As shown in figure 4.3.3, the new maximum number is 880000, it was 2000000.

In the **Price** attribute, there was the problem of null values and outliers. We fixed these problems by imputing the null values with the median and removing any record with a **Price** less than 5000 SAR. Removing the outliers was necessary because most

of the cars with a **Price** less than 5000 were new and luxury cars, so a **Price** less than 5000 did not seem reasonable.

| Min. | 1st Qu. | Median | Mean  | 3rd Qu. | Max.    |
|------|---------|--------|-------|---------|---------|
| 5000 | 41000   | 65000  | 94841 | 120000  | 1220000 |

Figure 4.3.4, A figure showing the new minimum and maximum numbers of the Price attribute

In the **Region** attribute, we had too many regions name or cities that refers to the same **Region** or city name. So, we decided to combine and rename those names into the names of 13 provinces of Saudi Arabia.

|          |           |                    |           |           |
|----------|-----------|--------------------|-----------|-----------|
| "riyadh" | "eastern" | "makkah"           | "madinah" | "aseer"   |
| "tabouk" | "jizan"   | "qassim"           | "al-baha" | "al-jouf" |
| "hail"   | "najran"  | "northern borders" |           |           |

Figure 4.3.5, A figure showing the new unique regions names

For the **Model** attribute, we had too many models in each brand referring to the same name, so we combined similar names.

|          |          |            |          |                |           |
|----------|----------|------------|----------|----------------|-----------|
| "innova" | "rush"   | "fortuner" | "hilux"  | "land cruiser" | "corolla" |
| "avalon" | "camry"  | "prado"    | "rav4"   | "fj"           | "yaris"   |
| "hiace"  | "avanza" | "cressida" | "previa" | "aurion"       |           |

Figure 4.3.6, Toyota brand models after cleaning

Next, we lowercase all the categorical attributes values, and also used the Standard Scaler function on the **Mileage** attribute. Standard Scaler is a function from the "sklearn" library, which is used to standardize features by removing the mean and scaling unit variance [2]. The main reason for choosing 'StandardScaler' is that it keeps the same distribution of data before scaling. The equation of the standard scaler is as follows:

$$z = \frac{x - \mu}{\sigma}$$

Equation 4.3-1, Standard Scaler Equation

Where Z is the new value after scaling, x is the value before scaling,  $\mu$  is the mean of the column/feature, and sigma is the standard deviation of the column/feature.

After that, we decided the need of adding another variable called "Age" which represents the car age using a feature extraction process from the year attribute. It is calculated by subtracting the 'Year' attribute, which is the year that car was made in, from the current year. The main goal of this attribute is to lower the values of the 'Year' attribute, which are considered high. For example, the 'Year' could be 2020 but the 'Age' is 3 and that helps in raising the accuracy of a model without using scalers.

Last but not least, we sliced the data into multiple datasets depending on the car brands. For example, all the records with a brand equal to Toyota will be on a separate dataset. The goal behind this is to analyze each brand with their car models separately so we can make a solid study with reliable and accurate results.

Finally, we transformed all the categorical attributes on all of the datasets into numeric values by using the `get dummies` function to make the data ready to use in the machine learning and deep learning models that we are going to build. The `get dummies` function is used to convert the categorical variables into dummy/indicator variables [3].

## 4.4 Conclusion

To conclude, in this chapter, we discussed what problems the dataset had like the null values and the outlier's problems or the low count of specific values in some attributes that may hurt the model accuracy. Then, we performed a data preprocessing operation and solved these problems, like imputing the null values and removing the outliers. Also, we prepared the data for deploying the model, by applying the Standard Scaler in the Mileage attribute and transforming all the categorical attributes values to numeric values. Now, after this chapter, the data is cleaned, organized, and prepared for deploying the model.

## 4.5 References

- [1] <https://pandas.pydata.org/>
- [2] [sklearn.preprocessing.StandardScaler — scikit-learn 1.1.3 documentation](#)
- [3] [pandas.get\\_dummies — pandas 1.5.1 documentation \(pydata.org\)](#)

## 5. Chapter 5: Model Building

### 5.1 Introduction

In the previous chapter, we preprocessed the data, discovered what problems it had, solved these problems, and prepared the data for model building and testing. In this chapter, the model-building phase will be discussed along with everything related to it. First, will discuss the experiment setup and what tools and libraries we are going to use. Then, we will mention how many models we are going to test, set up their initial parameters, and what is the selection criteria for each value in the parameters.

### 5.2 Experiments Setup and Tools

In this section of the chapter, we will discuss the experiment setup and tools and what we need to prepare before building the models and then testing it. Firstly, the IDE that we are going to use is Spyder IDE. Spyder is a free and open-source scientific environment written in Python, for Python, and designed by and for scientists, engineers, and data analysts. It features a unique combination of the advanced editing, analysis, debugging, and profiling functionality of a comprehensive development tool with the data exploration, interactive execution, deep inspection, and beautiful visualization capabilities of a scientific package.[1]

Secondly, as for the libraries we need before we start building the models, we need to import libraries starting from the sklearn library and its algorithms and Machine Learning models to the Keras library that we are going to need to build the Deep Learning model.

```
# To split the data into training and testing sets
from sklearn.model_selection import train_test_split
# Machine Learning Models
from sklearn.linear_model import LinearRegression
from sklearn.ensemble import RandomForestRegressor
from sklearn.tree import DecisionTreeRegressor
from sklearn.neighbors import KNeighborsRegressor
from sklearn.ensemble import GradientBoostingRegressor
from sklearn.svm import SVR
from sklearn.ensemble import ExtraTreesRegressor
from xgboost.sklearn import XGBRegressor
from catboost import CatBoostRegressor
# Deep Learning
from keras.models import Sequential
from keras.layers import Dense
from keras.wrappers.scikit_learn import KerasRegressor
import tensorflow as tf
# Metrics for evaluating the models
from sklearn.metrics import r2_score
from sklearn.metrics import mean_absolute_error
from sklearn.metrics import mean_absolute_percentage_error
import timeit
```

Figure 5.2.1, A screenshot from Spyder IDE Shows the libraries imported.

Finally, to start building our models we need to split the data into a training set and a test set. We discussed before in chapter 4 that we sliced the data into 9 datasets based on the car brand so that we can predict the price for each car brand independently. So, each dataset is going to be split into 80% for the training set and 20% for the testing set, and each dataset will be tested independently.

### 5.3 Initial Parameters and Selection Criteria

In this part of the chapter, we are going to discuss the models we are going to use, set up their initial parameters, and what is the selection criteria for the values in the parameters.

After searching for regressors models in the Scikit-Learn documentation and other verified documentation, we selected 9 machine learning models to test and predict the price of used cars in Saudi Arabia. The models we selected are **Random Forest Regressor**, **Decision Tree Regressor**, **K Nearest Neighbors Regressor**, **Extreme Gradient Boosting Regressor**, **Gradient Boosting Regressor**, **Linear Regression**, **Extra Trees Regressor**, **Cat Boost Regressor**, and **Support Vector Regressor**. Also, we are going to build a **Deep Learning Model** that we will discuss its parameter later in this section with the other machine learning model parameters.

The selection criteria for most of the values in the parameters that we are going to discuss below were based on what is the value that can give us the best accuracy. Meaning we tried many values in each parameter below and we selected the values that gave the best accuracy.

– **Random Forest, Extra Trees, and Decision Tree Regressors.**[4]

```
RandomForestRegressor(n_estimators = 85, criterion = 'squared_error')  
ExtraTreesRegressor(n_estimators = 47, criterion = 'squared_error')  
DecisionTreeRegressor(criterion = 'squared_error', max_depth=15)
```

*Figure 5.31., The three tree model parameters*

**n\_estimators:** Specifies the number of trees in the model. The number of trees selected for each of the two tree models (**Random Forest & Extra Trees**) was based on the highest accuracy after trying many other numbers of trees.

**criterion:** Function to measure the quality of a split. The "squared error" function was selected for all three models. The function "squared error" is the mean squared error based on the accuracy. Other functions were tested like "absolute error" and "poisson" but it didn't give better accuracy than "squared error".

**max\_depth:** Specifies the maximum depth of the tree. This parameter as shown in figure 5.3.1 above is available in the **Decision tree model** and not the other models. The number 15 was the ideal number.



### – Gradient Boosting and Extreme Gradient Boosting Regressors.[2]

```
XGBRegressor(n_estimators=495, max_depth=7, eta=0.1, subsample=0.9, colsample_bytree=0.7)
GradientBoostingRegressor(n_estimators=600)
```

*Figure 5.3.2, The two Gradient Boosting Regressors parameters*

**n\_estimators:** Number of boosting stages. As for the two tree models before, here the number of boosting stages that gave the best accuracy was selected for each of the two models.

**max\_depth:** Maximum depth of the individual regression estimators, which limits the number of nodes in the tree. Here the best value depends on the interaction of the input variables. The best value for our case was 7.

**eta:** Alias for learning rate and it is used to step size shrinkage used in the update to prevent overfitting. The number 0.1 was selected.

**subsample:** The ratio of the training instances is used to prevent overfitting. The number 0.9 was selected meaning that **XGBOOST** will randomly sample 0.9 of the training data before growing trees.

**Colsample\_bytree:** Subsample ratio of columns when constructing each tree. The value selected was 0.7.

### – Cat Boost Regressor.[3]

```
CatBoostRegressor(iterations= 108)
```

*Figure 5.3.3, Cat Boost Regressor parameters*

**iterations:** Number of iterations. Here we only used this one parameter and selected the number that gave us the best accuracy.

### – K Nearest Neighbors.[4]

```
KNeighborsRegressor(n_neighbors=4, algorithm='auto', weights='distance', metric='manhattan')
```

*Figure 5.3.4, K Nearest Neighbors parameters*

**n\_neighbors:** Number of neighbors. In the objectives in chapter 1, we stated that we are going to analyze each car brand individually. Meaning that we are going to split the data into many sets based on the car brand and test the models on each set individually. So, the ideal number of neighbors here is different for each set, but most of them had the value of 4 neighbors as the ideal value.

**algorithm:** Algorithm used to compute. The "ball\_tree" and "kd\_tree" algorithms were tested but putting the value as "auto" gave the best results.

**weights:** Weight function used in prediction. The "uniform" function was tested but the "distance" function gave better results. The "distance" function means the weights points by the inverse of their distance, giving us the best results.

**metric:** metric used for distance computation. The "euclidean" and "minkowski" metrics were tested but the "manhattan" metric gave the best results.

#### – Support Vector Regressor and Linear Regression.[4]

```
SVR(kernel = 'poly',C=300,gamma=0.15)
LinearRegression()
```

*Figure 5.3.5, Support Vector Regression and Linear Regression parameters*

**kernal:** Kernal type, is used to precompute the kernal matrix. The "rbf", "linear", and "sigmoid" kernalns were tested but none of them gave a better accuracy than the "poly" kernal.

**C:** Regularization parameter, the strength of the regularization is inversely proportional to C. The value here was selected based on the highest accuracy.

**gamma:** Kernal coefficient. The "scale" value was tested and this value means that when the value is passed then it uses  $1 / (n\_features * X.var())$  as the value of gamma but it didn't give good accuracy. So, instead, we inserted a value manually which is the number 0.15 that gave the best accuracy.

For the **Linear Regression** model, we tried adding parameters for the model, but the accuracy did not increase, instead, it decreased so, leaving the model with no parameters gave us better accuracy. Also, in general, this model does not perform very well with this type of project as we searched.

#### – Deep Learning.

```
# create model
model = Sequential()
model.add(Dense(10, input_dim=33, activation='relu'))
model.add(Dense(30, activation='relu'))
model.add(Dense(40, activation='relu'))
model.add(Dense(1))
# Compile model
model.compile(optimizer = 'adam', loss = 'mean_squared_error',
              metrics = ['mae'])
# Fit the model
model.fit(x_train, y_train, validation_data=(x_test,y_test), epochs=250, batch_size=32)
```

*Figure 5.3.6, Deep Learning model parameters*

For the Deep Learning model parameters, we built the model by creating **one input layer** and specified its **input dimension** (the **input dimension** is different for each dataset) and the number of units is 10 (which is the first number as we can see in the figure, second line). **Unit** is a positive integer and it represents the dimensionality of

the output space. Also, we added **2 hidden layers**, the first hidden layer has the number 30 as the unit number, and the second hidden has the number 40 as the unit number. Both layers have the same activation function which is Relu. **Relu** is the rectified linear unit function. More on that, we added **one output layer** as we can see in the figure. Then, we compiled the model, the **optimizer** here was **adam**. An **optimizer** is required for compiling a Keras model, and **adam** is a stochastic gradient descent method that is based on adaptive estimation of first-order and second-order moments. For the loss function and the metric for evaluating the model, we selected mean squared error. Finally, we fitted the model with 250 as the number of **epochs** or iterations and 32 as the **batch size**. Batch size is the number of samples that will be propagated through the network.[5]

## 5.4 Conclusion

In conclusion, in this chapter, we discussed building many models like the Random Forest, Decision Tree, ...etc. and how we set the experiment environment for the model testing by splitting the data into 20% for testing and 80% for training. Also, we specified what libraries we imported to completely set up the experiment environment. Then, we discussed the parameters for each model and explained that the selection criteria for most of the values in the parameters were based on accuracy.

## 5.5 References

- [1][Home — Spyder IDE \(spyder-ide.org\)](https://spyder-ide.org/)
- [2][XGBoost Parameters — xgboost 1.7.3 documentation](#)
- [3][CatBoost](#)
- [4][scikit-learn: machine learning in Python — scikit-learn 1.2.1 documentation](#)
- [5] [Keras: the Python deep learning API](#)

## 6. Chapter 6: Results and Discussions

### 6.1 Introduction

In the last chapter, we discussed and provided what models we are going to test, their building, their parameters, selection criteria for the values in the parameters, and the experiment setup and tools. In this chapter, we are going to discuss the results of the models in detail for all the datasets. Also, we will explain the math behind the evaluation metrics that we are going to use to evaluate the performance of the models. Then, in the end, we will discuss the results of the best model and explain why we see it as the best-performing model among all the others.

### 6.2 Performance Evaluation Metrics

In this section of the chapter, we will explain the math behind the evaluation metrics we selected to evaluate the performance of the models. The evaluation metrics selected were R Squared, Mean Absolute Error, and Mean Absolute Error Percentage. R Squared represents the accuracy of the model in percentage, Mean Absolute Error represents the error in the prediction in the form of the amount of the price, and Mean Absolute Error Percentage represents the same value in the Mean Absolute Error but in percentage.

- R Squared Equation: -

$$R^2 = 1 - \frac{\text{Sum Squared Regression}}{\text{Total Sum of Squares}} = 1 - \frac{\sum (y_i - \hat{y}_i)^2}{\sum (y_i - \bar{y})^2}$$

*Equation 6.2-1, R Squared Equation*

*The sum squared regression in the equation above is the sum of the residuals squared (residual for each observation is the difference between predicted values of y and observed values of y), and the total sum of squares is the sum of the distance the data is away from the mean all squared. As it is a percentage it will take values between 00 and 11.[1]*

- Mean Absolute Error Equation: -

$$MAE = \frac{1}{n} * \sum |x_i - x|$$

*Equation 6.2-2, Mean Absolute Error Equation*

Mean Absolute Error calculates the average difference between the calculated values and actual values. As we see in the equation above,  $x_i$  is the calculated value for the  $i$ th observation,  $x$  is the actual value for the  $i$ th observation, and  $n$  is the total number of observations.[2]

Mean Absolute Percentage Error: -

$$MAPE = \frac{1}{n} * \sum_{t=1}^n \left| \frac{A_t - F_t}{A_t} \right|$$

*Equation 6.2-3, Mean Absolute Percentage Error Equation*

Where,  $n$  is the number of fitted points,  $A_t$  is the actual value, and  $F_t$  is the forecast value. [3]

## 6.3 Experiments Results

In this part of the chapter, we will discuss two aspects of the results of the models we tested on the data and, we will provide visualizations of the accuracies to be clearer to understand. As we stated before in the previous chapters, we sliced the data into different sets based on the car brand. So, the first aspect we will discuss is the results of the models on all the data without separating them into different sets based on the car brand. Next, the second aspect we will discuss is the results of each set we separated based on the car brand.

**The first aspect is the results of the models on all the data without separating it: -**

We tested 9 different Machine Learning models and a Deep Learning model on all the data.

| All Data Results          |       |          |       |
|---------------------------|-------|----------|-------|
| Model                     | MAPE  | MAE      | R2    |
| DecisionTreeRegressor     | 0.203 | 13878.37 | 0.911 |
| XGBRegressor              | 0.154 | 10533.74 | 0.96  |
| GradientBoostingRegressor | 0.182 | 12250.81 | 0.952 |
| RandomForestRegressor     | 0.155 | 10956.78 | 0.948 |
| LinearRegression          | 0.435 | 21845.84 | 0.808 |
| ExtraTreesRegressor       | 0.147 | 10188.72 | 0.957 |
| CatBoostRegressor         | 0.178 | 12184.98 | 0.951 |
| SVR                       | 0.269 | 19351.96 | 0.752 |
| KNN                       | 0.166 | 11713.79 | 0.938 |
| Deep Learning             | 0.166 | 11501.99 | 0.955 |

*Table 6.3-1, A table shows the detailed results of the 10 models on the data.*

We can see from table 6.3-1 that the best-performing model on all the data without separating it was the Extra Tree Regressor. It is the best performing model based on the results of the three metrics. Below we provide visualizations of the accuracies of the models, the visualizations provided for more clarity of the results.

The results of the ten models used in the below graphs are shown as ‘actual vs. predicted’ graphs that indicate the accuracies of the predictions made on the test data. To help interpret the meaning of these graphs, you can use the following statement: The more the data points are near the line, the more the accuracy score is.



Figure 6.3.1, Visualization of the  $R^2$  accuracy for the **Decision Tree Regressor** model

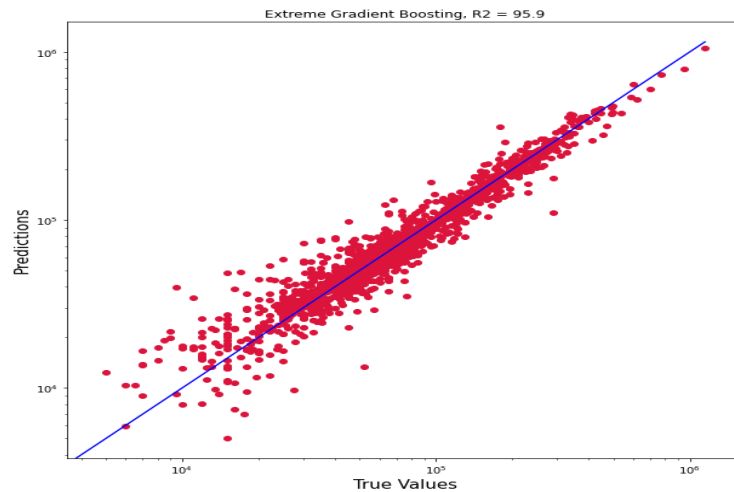


Figure 6.3.2, Visualization of the  $R^2$  accuracy for the **Extreme Gradient Boosting Regressor** model

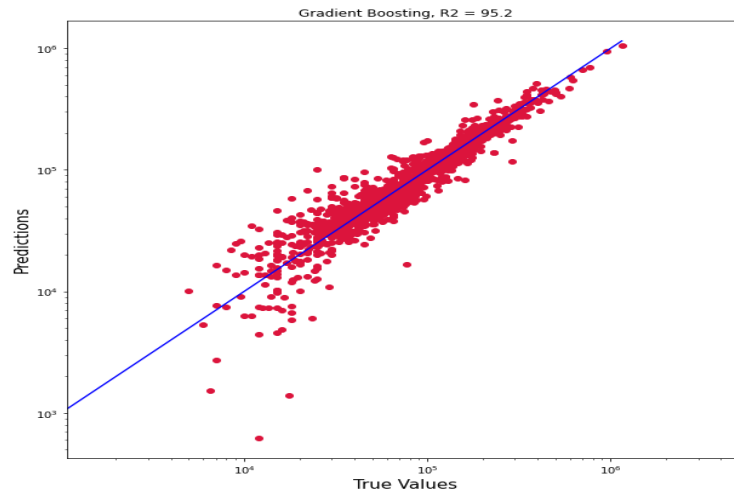


Figure 6.3.3, Visualization of the  $R^2$  accuracy for the **Gradient Boosting Regressor** model

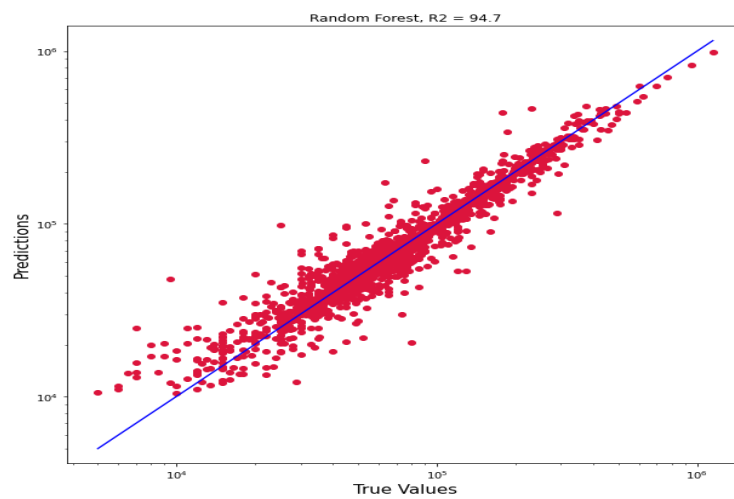


Figure 6.3.4, Visualization of the  $R^2$  accuracy for the **Random Forest Regressor** model

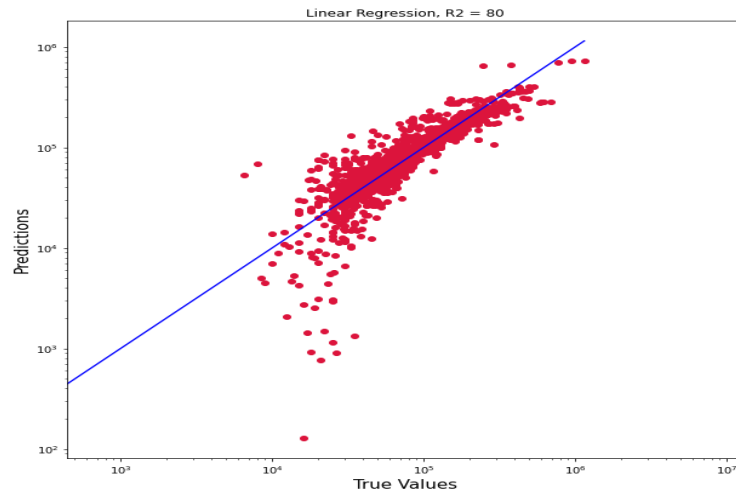


Figure 6.3.5, Visualization of the  $R^2$  accuracy for the **Linear Regression** model

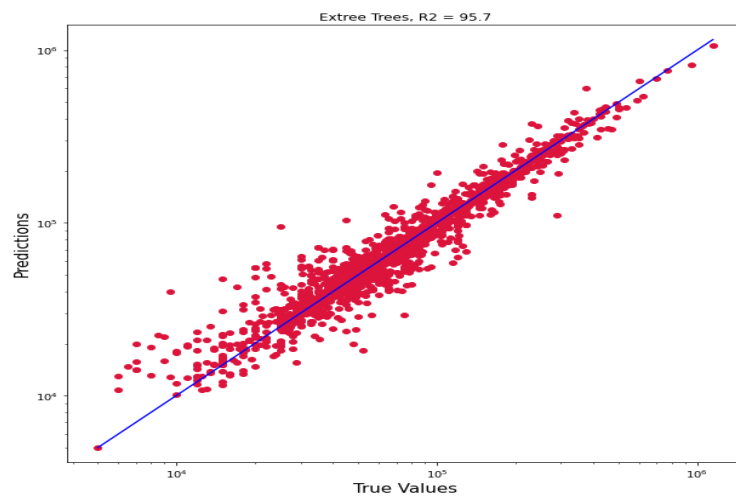


Figure 6.3.6, Visualization of the  $R^2$  accuracy for the **Extra Trees Regressor** model

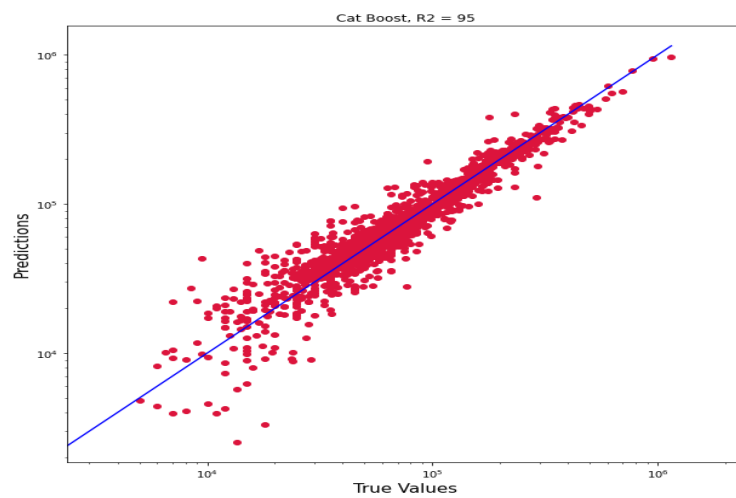


Figure 6.3.7, Visualization of the  $R^2$  accuracy for the **Cat Boost Regressor** model



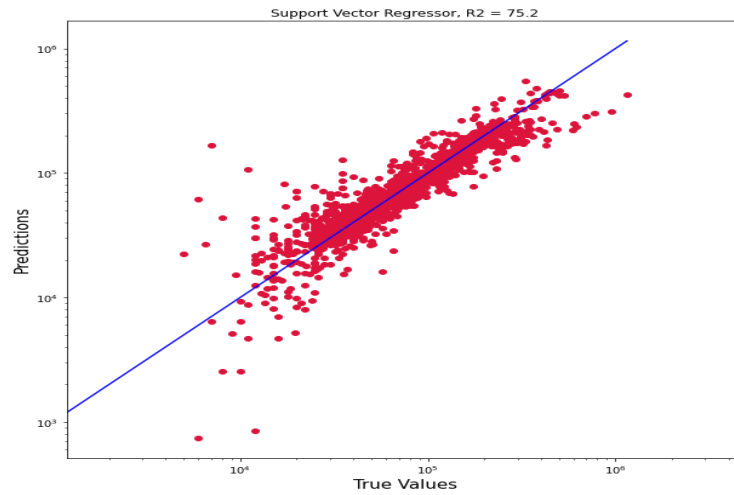


Figure 6.3.8, Visualization of the  $R^2$  accuracy for the *Support Vector Regressor* model

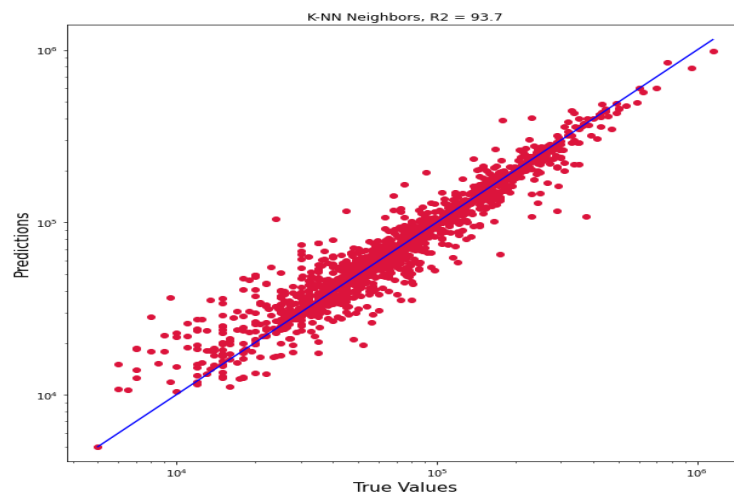


Figure 6.3.9, Visualization of the  $R^2$  accuracy for the *K Nearest Neighbours Regressor* model

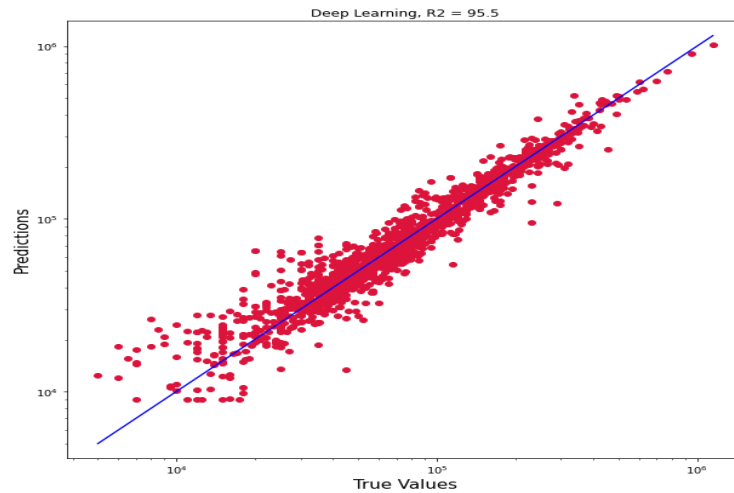


Figure 6.3.10, Visualization of the  $R^2$  accuracy for the **Deep Learning** model

**The second aspect, the results of the models on all the separated sets that are separated based on the car brand: -**

As we did above, we tested 9 different Machine Learning models and a Deep Learning model on all 9 separated sets, each set individually. The colored green rows on the below accuracy tables refer to the best performing model on the set. The model is identified as the best based on the three metrics provided and not only the R Squared accuracy.

| Toyota                    |       |           |       |
|---------------------------|-------|-----------|-------|
| Model                     | MAPE  | MAE       | R2    |
| DecisionTreeRegressor     | 0.194 | 11774.163 | 0.832 |
| XGBRegressor              | 0.157 | 8161.114  | 0.957 |
| GradientBoostingRegressor | 0.171 | 9305.714  | 0.951 |
| RandomForestRegressor     | 0.163 | 9085.19   | 0.943 |
| LinearRegression          | 0.475 | 18818.374 | 0.785 |
| ExtraTreesRegressor       | 0.155 | 8537.268  | 0.951 |
| CatBoostRegressor         | 0.167 | 9306.745  | 0.953 |
| SVR                       | 0.275 | 13526.966 | 0.858 |
| KNN                       | 0.168 | 9473.622  | 0.935 |
| Deep Learning             | 0.181 | 10577.711 | 0.933 |

Table 6.3-2, A table shows the detailed results of the 10 models on the Toyota cars data

| Nissan                    |       |          |       |
|---------------------------|-------|----------|-------|
| Model                     | MAPE  | MAE      | R2    |
| DecisionTreeRegressor     | 0.209 | 12285.03 | 0.902 |
| XGBRegressor              | 0.217 | 11636.92 | 0.918 |
| GradientBoostingRegressor | 0.198 | 10346.15 | 0.943 |
| RandomForestRegressor     | 0.18  | 11178.43 | 0.921 |
| LinearRegression          | 0.505 | 22897.21 | 0.776 |
| ExtraTreesRegressor       | 0.17  | 10479.24 | 0.922 |
| CatBoostRegressor         | 0.194 | 10719.96 | 0.922 |
| SVR                       | 0.329 | 23971.67 | 0.734 |
| KNN                       | 0.18  | 10954.82 | 0.923 |
| Deep Learning             | 0.234 | 14262.83 | 0.908 |

Table 6.3-3, A table shows the detailed results of the 10 models on the Nissan cars data

| GMC                       |       |           |       |
|---------------------------|-------|-----------|-------|
| Model                     | MAPE  | MAE       | R2    |
| DecisionTreeRegressor     | 0.195 | 13861.61  | 0.872 |
| XGBRegressor              | 0.213 | 12346.389 | 0.913 |
| GradientBoostingRegressor | 0.223 | 12211.238 | 0.918 |
| RandomForestRegressor     | 0.203 | 11621.201 | 0.929 |
| LinearRegression          | 0.752 | 21220.615 | 0.721 |
| ExtraTreesRegressor       | 0.232 | 11246.079 | 0.921 |
| CatBoostRegressor         | 0.212 | 11483.033 | 0.926 |
| SVR                       | 0.677 | 19784.356 | 0.706 |
| KNN                       | 0.232 | 13606.847 | 0.865 |
| Deep Learning             | 0.277 | 17193.511 | 0.827 |

Table 6.3-4, Table 6.3 3, A table shows the detailed results of the 10 models on the GMC cars data

| Mercedes                  |       |          |       |
|---------------------------|-------|----------|-------|
| Model                     | MAPE  | MAE      | R2    |
| DecisionTreeRegressor     | 0.142 | 28891.21 | 0.854 |
| XGBRegressor              | 0.117 | 24012.21 | 0.921 |
| GradientBoostingRegressor | 0.113 | 26365.57 | 0.867 |
| RandomForestRegressor     | 0.112 | 25654.03 | 0.905 |
| LinearRegression          | 0.324 | 40709.7  | 0.801 |
| ExtraTreesRegressor       | 0.118 | 25061.91 | 0.914 |
| CatBoostRegressor         | 0.111 | 24105.17 | 0.935 |
| SVR                       | 0.409 | 64088.95 | 0.302 |
| KNN                       | 0.115 | 27498.07 | 0.875 |
| Deep Learning             | 0.21  | 48097.36 | 0.713 |

Table 6.3-5, A table shows the detailed results of the 10 models on the Mercedes cars data

| Kia                       |       |          |       |
|---------------------------|-------|----------|-------|
| Model                     | MAPE  | MAE      | R2    |
| DecisionTreeRegressor     | 0.145 | 6166.667 | 0.561 |
| XGBRegressor              | 0.123 | 5075.906 | 0.77  |
| GradientBoostingRegressor | 0.128 | 5490.787 | 0.71  |
| RandomForestRegressor     | 0.14  | 5518.373 | 0.705 |
| LinearRegression          | 0.157 | 6523.696 | 0.713 |
| ExtraTreesRegressor       | 0.151 | 5798.679 | 0.669 |
| CatBoostRegressor         | 0.128 | 5439.948 | 0.787 |
| SVR                       | 0.21  | 6998.697 | 0.644 |
| KNN                       | 0.156 | 5829.161 | 0.675 |
| Deep Learning             | 0.192 | 7933.693 | 0.58  |

Table 6.3-6, A table shows the detailed results of the 10 models on the Kia cars data

| Chevrolet                 |       |          |       |
|---------------------------|-------|----------|-------|
| Model                     | MAPE  | MAE      | R2    |
| DecisionTreeRegressor     | 0.192 | 10274.96 | 0.892 |
| XGBRegressor              | 0.194 | 9753.06  | 0.934 |
| GradientBoostingRegressor | 0.176 | 9712.395 | 0.944 |
| RandomForestRegressor     | 0.171 | 9232.786 | 0.939 |
| LinearRegression          | 0.459 | 17773.75 | 0.806 |
| ExtraTreesRegressor       | 0.16  | 8547.858 | 0.946 |
| CatBoostRegressor         | 0.207 | 10090.66 | 0.934 |
| SVR                       | 0.329 | 13861.92 | 0.835 |
| KNN                       | 0.187 | 9639.102 | 0.928 |
| Deep Learning             | 0.197 | 13310.6  | 0.888 |

Table 6.3-7, A table shows the detailed results of the 10 models on the Chevrolet cars data

| Hyundai                   |       |          |       |
|---------------------------|-------|----------|-------|
| Model                     | MAPE  | MAE      | R2    |
| DecisionTreeRegressor     | 0.141 | 6261.976 | 0.75  |
| XGBRegressor              | 0.111 | 4853.693 | 0.883 |
| GradientBoostingRegressor | 0.117 | 5167.246 | 0.873 |
| RandomForestRegressor     | 0.115 | 4948.836 | 0.87  |
| LinearRegression          | 0.177 | 6437.455 | 0.768 |
| ExtraTreesRegressor       | 0.117 | 5205.528 | 0.851 |
| CatBoostRegressor         | 0.117 | 4995.578 | 0.885 |
| SVR                       | 0.174 | 6916.571 | 0.461 |
| KNN                       | 0.119 | 5133.629 | 0.838 |
| Deep Learning             | 0.143 | 6058.936 | 0.768 |

Table 6.3-8, A table shows the detailed results of the 10 models on the Hyundai cars data

| Ford                      |       |          |       |
|---------------------------|-------|----------|-------|
| Model                     | MAPE  | MAE      | R2    |
| DecisionTreeRegressor     | 0.209 | 11109.4  | 0.729 |
| XGBRegressor              | 0.197 | 9543.918 | 0.85  |
| GradientBoostingRegressor | 0.187 | 9387.466 | 0.862 |
| RandomForestRegressor     | 0.197 | 9626.716 | 0.838 |
| LinearRegression          | 0.342 | 15023.62 | 0.719 |
| ExtraTreesRegressor       | 0.188 | 8651.136 | 0.871 |
| CatBoostRegressor         | 0.196 | 9706.785 | 0.871 |
| SVR                       | 0.275 | 12954.74 | 0.693 |
| KNN                       | 0.19  | 10079.9  | 0.805 |
| Deep Learning             | 0.241 | 12730.74 | 0.791 |

Table 6.3-9, A table shows the detailed results of the 10 models on the Ford cars data

| Lexus                     |       |           |       |
|---------------------------|-------|-----------|-------|
| Model                     | MAPE  | MAE       | R2    |
| DecisionTreeRegressor     | 0.121 | 24287.086 | 0.927 |
| XGBRegressor              | 0.113 | 21481.772 | 0.925 |
| GradientBoostingRegressor | 0.113 | 21777.009 | 0.923 |
| RandomForestRegressor     | 0.118 | 22299.028 | 0.916 |
| LinearRegression          | 0.187 | 38631.71  | 0.836 |
| ExtraTreesRegressor       | 0.12  | 22483.718 | 0.921 |
| CatBoostRegressor         | 0.118 | 24444.967 | 0.918 |
| SVR                       | 0.161 | 38224.315 | 0.709 |
| KNN                       | 0.131 | 26916.692 | 0.901 |
| Deep Learning             | 0.197 | 44667.374 | 0.723 |

Table 6.3-10, A table shows the detailed results of the 10 models on the Lexus cars data

| Average Results of the 9 Car Brands Sets |              |             |            |
|--|--------------|-------------|------------|
| Model                                    | Average MAPE | Average MAE | Average R2 |
| DecisionTreeRegressor                    | 0.172        | 13879.122   | 0.813      |
| XGBRegressor                             | 0.160        | 11873.888   | 0.897      |
| GradientBoostingRegressor                | 0.158        | 12195.952   | 0.888      |
| RandomForestRegressor                    | 0.155        | 12129.399   | 0.885      |
| LinearRegression                         | 0.375        | 20892.903   | 0.769      |
| ExtraTreesRegressor                      | 0.157        | 11779.046   | 0.885      |
| CatBoostRegressor                        | 0.161        | 12254.761   | 0.903      |
| SVR                                      | 0.315        | 22258.688   | 0.660      |
| KNN                                      | 0.164        | 13236.872   | 0.861      |
| Deep Learning                            | 0.208        | 19425.861   | 0.792      |

Table 6.3-11, A table shows the detailed average results of the 10 models on all the 9 separated car brand sets data

## 6.4 Discussion

In general, after we finally finished the model (which is the core of this paper), performed the tests, and got the results. We believe that the outcome results of this paper have met our expectations. The main objective of this paper was to analyze the market of used cars in Saudi Arabia and solve the problem of setting wrong estimations of prices for the cars. The paper has achieved this objective and the other objectives that we have stated before in chapter 1. We collected data about used cars in Saudi Arabia from several famous websites in the country, cleaned this data, explored it, analyze it, and tested different models on it to predict the prices. Also, we learned so much information about the used cars market in Saudi Arabia and realized its importance today and its importance in the future. 10 different models or algorithms were tested to expand our view more on the results, and one of the models was a Deep Learning model. The best performing model in our view was the **Extra Trees Regressor**. We believe that this is the best model because of the high R squared accuracy, the low Mean Absolute error, and the low Mean Absolute Percentage Error. The model performed well on all the data and all the 9 separated car brands datasets. Another model that performed well is the **Extreme Gradient Boosting Regressor**. The model had pretty much great results as the **Extra Trees Regressor** model, but it hadn't had low values on the Mean Absolute Error and Mean Absolute Percentage Error as the **Extra Trees Regressor** had. As an outcome, we see the Extra Trees Regressor model as the best model, and we also see and believe that the model we built can give realistic results and be used as an assistance for any individual who wants to estimate the price of a used car in Saudi Arabia.

## 6.5 Conclusion

In conclusion of this chapter, the equations of the evaluation metrics used to evaluate the results were explained, and the results of each model on all the data and all the 9 separated car brands datasets were provided. The results in general were good on most of the models, but the best-performing model was the **Extra Trees Regressor**. The model had a 95.7% R Squared accuracy on all the data and 88.5% R Squared average accuracy on all the 9 car brands separated datasets. The lowest-performing model was the **Support Vector Regressor**. Now, as the model is done, the model is ready and can be deployed to the web application that will help end users in estimating used car prices in Saudi Arabia.

## 6.6 References

- [1] [Numeracy, Maths and Statistics - Academic Skills Kit \(ncl.ac.uk\)](https://www.ncl.ac.uk/academic-skills-kit/)
- [2] [How to Calculate Mean Absolute Error in Python? - GeeksforGeeks](https://www.geeksforgeeks.org/how-to-calculate-mean-absolute-error-in-python/)
- [3] <https://www.statisticshowto.com/mean-absolute-percentage-error-mape/>

## 7. Chapter 7: Conclusion and Future Work

### 7.1 Introduction

In this chapter, we will conclude our work in this project by choosing the best-performing algorithm out of the aforementioned ten. Furthermore, the dashboard that describes the used car market in Saudi Arabia will be discussed thoroughly. Additionally, the web solution that the machine learning model is deployed on will be presented. Moreover, the difficulties and limitations that the project posed will be discussed. Finally, some ideas for future work will also be presented.

### 7.2 Conclusion

In conclusion, after testing and experimenting with several machine and deep learning algorithms, the chosen algorithm to be deployed on the web solution is the Extra Trees Regressor with  $n_{\text{estimators}}$  of 47 and chosen criterion of 'squared\_error'. The reason for choosing the hyperparameters of the algorithm was that these hyperparameters performed the best empirically. The evaluation criteria chosen were R-squared score or coefficient of determination, Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE), in which the Extra Trees Regressor achieved a score of 95.7%, 10,188.7, and 14.7% respectively.

### 7.3 The Interactive Dashboard

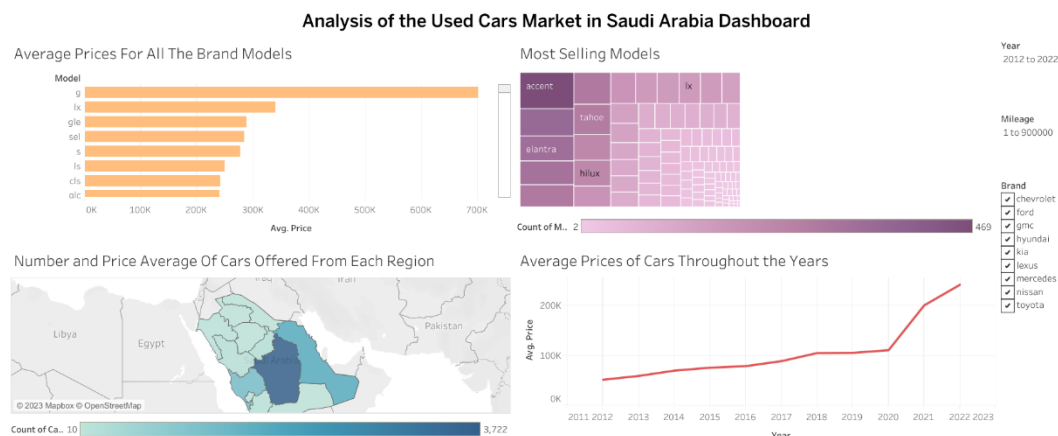


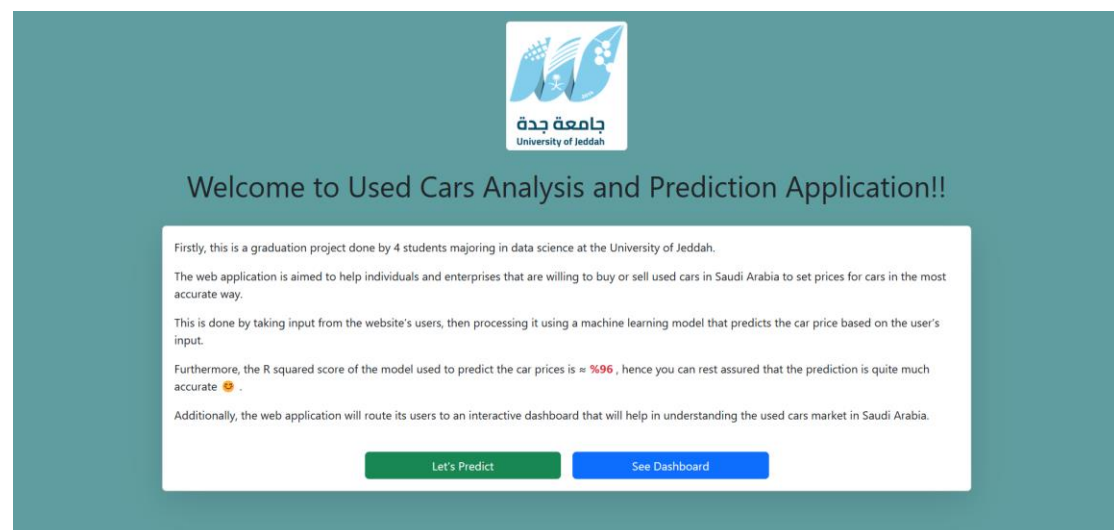
Figure 7.3.1, The Interactive Dashboard

As shown in Figure 7.3.1, the dashboard was created into four quadrants, and the position of each visualization represents its degree of emphasis. From left to right, the first visualization is a horizontal bar chart representing the Models of the car with their average prices as the most emphasized visualization. Moving on, the second visualization is a tree map that shows the number of models available from each brand, this represents high to neutral emphasis. The third visualization is a map that shows the

number and price averages of cars based on the region they were offered from; this visualization represents low to neutral emphasis. The last visualization is a line chart that shows the average prices of cars over the years; this visualization represents the least emphasis on the dashboard. Furthermore, on the far right of the dashboard, the brand, year, and mileage of the cars, have been added as filters. Finally, the dashboard was uploaded to Tableau Public, which is a service that allows its users to upload their Tableau-made visualizations to the web. To see the dashboard, [click here](#).

## 7.4 The Web Solution

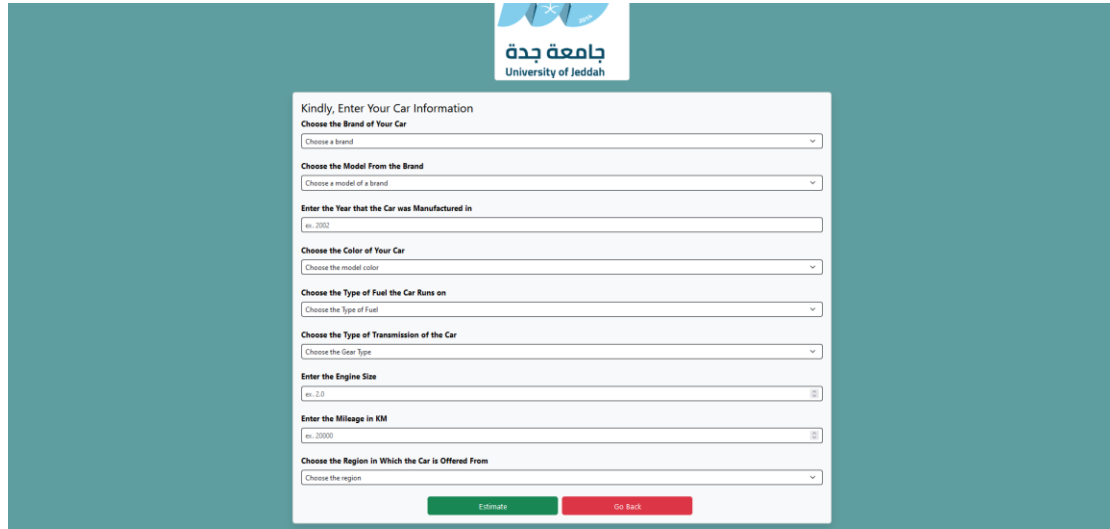
To help stakeholders of the project benefit from it, a web solution was made to deploy the machine learning model and the dashboard on. The application was made using multiple libraries in Python, such as Bootstrap, and Django. Additionally, Heroku was used as a host for the web solution to make it public. The web solution is a two-page website that will be discussed in detail later.



*Figure 7.4.1, The First Page of The Web-solution*

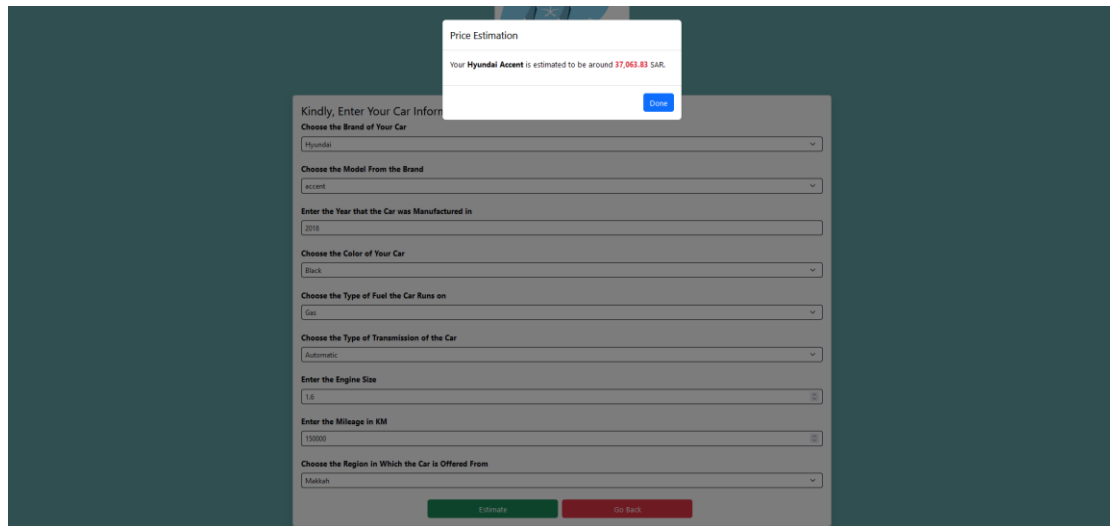
As shown in Figure 7.4.1, the first page is a welcoming page for the website's visitors that will inform them of the function of the website. Furthermore, two buttons were added that prompt the user to different pages. The green button will prompt the user to the second page of the website which will prompt the user to the prediction page. The blue button will prompt the user to the aforementioned dashboard that is uploaded to Tableau Public. To see the page, [click here](#).





The screenshot shows a web form titled "Kindly, Enter Your Car Information" with the University of Jeddah logo at the top. The form contains several input fields and dropdown menus: "Choose the Brand of Your Car" (dropdown), "Choose the Model From the Brand" (dropdown), "Enter the Year that the Car was Manufactured in" (text input with "ex. 2002"), "Choose the Color of Your Car" (dropdown), "Choose the Type of Fuel the Car Runs on" (dropdown), "Choose the Type of Transmission of the Car" (dropdown), "Enter the Engine Size" (text input with "ex. 2.0"), "Enter the Mileage in KM" (text input with "ex. 20000"), and "Choose the Region in Which the Car is Offered From" (dropdown). At the bottom are two buttons: "Estimate" (green) and "Go Back" (red).

*Figure 7.4.2, The Prediction Page*



The screenshot shows the same form as Figure 7.4.2, but with a "Price Estimation" pop-up window. The pop-up displays: "Your Hyundai Accent is estimated to be around 37,063.83 SAR." with a "Close" button. The form fields are filled with: Brand: Hyundai, Model: accent, Year: 2018, Color: Black, Fuel: Gas, Transmission: Automatic, Engine Size: 1.6, Mileage: 15000, and Region: Makkah. The "Estimate" and "Go Back" buttons are still visible at the bottom.

*Figure 7.4.3, A Sample Output for a Prediction*

As shown in Figure 7.4.2, the page will ask the user to enter the car's to be estimated information. After the user fills all the fields and clicks on 'Estimate', the information would go to the back end as an input to the machine learning model, consequently, returning a price prediction that will be shown in Figure 7.4.3.

## 7.5 Difficulties and Limitations

As with any project, this project posed a lot of difficulties and limitations. Many of these limitations were solvable, while others were not. The main two limitations and difficulties we had was when worked on areas that are not related to our major and field of study. Those two limitations or difficulties were working on web scrapping the data from several sites and working on building the web application for the end users. Web scrapping the data was a difficult task for us because we didn't web-scrapped data before. So, this task took us time to learn and a lot of time to apply what we learned and scrap the data. For the web application, the web application was built using python, but we did not build or work on web applications before, so this task took time also. Another difficulty or limitation we had, was preprocessing the data and the size of the data. The data had too many problems, like missing values and values that were inserted by users and had wrong or unclear values. Those problems took a lot of time to solve. Moreover, the data collected had only 8103 rows and a limited number of car brands. We tried to scrap more data, but we didn't find websites that had reliable data as the websites we scrapped from them before.

## 7.6 Future Work

In the future, we plan to work more on the models we built, especially the Deep Learning model. As that, the Deep Learning model accuracy was not satisfying and needs more work and tuning to its parameters and layers. Also, we plan to find other reliable websites in Saudi Arabia and web scrap more data from them and apply more advanced web scrapping and preprocessing techniques so that the results can be more and more accurate. Additionally, we will try adding more features to the web application so that the end users can benefit more from it. In the end, working on this project has benefited us in a lot of areas and we will work on it in the future and try to improve it more.