**Linear Regression**

LINEAR REGRESSION MODEL

1. Loading Data

2. Handling Missing Values

3. Data Transformation

**Step 1:**

**Loading Data**

It consists of importing necessary Python libraries such as (pandas, numpy, matplotlib, seaborn) and loading the dataset named **pakwheels\_used\_cars.csv**. The dataset contains **77,237** entries and **14 columns**, including features like make ,model ,year ,mileage ,cc ,price ,registered ,fuel type etc. after running the cells it displays the **first few rows** and the dataset shape.

**Handling Missing Values:**

The cells Counts missing values per column and per row. And according to the stats:

There are number of missing values in columns such as assembly (**53,261** missing), body (**8,865** missing), and year (**4,721** missing).

Two approaches can be used :

**Dropping rows:**

Dropping rows means if any single piece of information is missing for an item (like a car or a person), you delete all the information about that and it can only be best when you have a HUGE amount of data (millions of forms) and only a tiny number of them have blanks (like 2 or 3). Throwing them out won't make a difference.

**Filling rows:**

Filling rows means you keep all your items, and for any missing information, you fill in the blank with a logical guess based on the data you *do* have.it is best when you can’t afford to lose data,

**Data Transformation** is the process in which we take the raw, messy data and prepare them so the model can understand and use it properly.

**Encoding & Scaling:**

**Label Encoding**

Label Encoding is simply assigning a unique number to each category in a column.

**Before and After applying encoding Example:**

|  |  |  |
| --- | --- | --- |
| **fuel\_type** |  | **fuel\_type (Encoded)** |
| Petrol |  | 0 |
| Hybrid |  | 2 |
| Petrol |  | 0 |
| Diesel |  | 1 |

**Purpose of usage of this in the document:**  
The car dataset had many columns with words like make, model, ad\_city, etc. To use them in the linear regression model, they had to be converted into numbers. That is why it was used.

**Standardization** fixes this by using features like **StandardScaler** so they’re on the same scale.

**Data Splitting**

Data splitting is the process of dividing your dataset into two separate pieces: a **training set**  80% and a **testing set 20%.**

**Exploratory Data Analysis (EDA):**

It shows basic statistics for each column, For example:

Price: The mean price is approximately 3.85 million PKR.

Mileage: The average mileage is 91,296 km.

Engine Capacity: The mean engine capacity is 1406cc.

**Correlation** tells you relationship between features. This is measured with a number between -1 and +1.

Manual cars tend to be cheaper than Automatic cars in this dataset.

**Visulization:**

For visualization we have used different things like:

* Histograms
* Scatter plots
* Box plots
* Correlation heatmap

**FINAL REPORT :**

Before we could build a tool to predict car prices, we first had to do three things:

1. Explore: Look through the raw data to see what we have.
2. Clean: Fix all the problems, like missing information and messy data.
3. Analyze: Play detective and find the hidden patterns and clues in the clean data.

**CLASSIFICATION**

**CLASSIFICATION MODEL**

**Data Cleaning and Preparation**

* **Loading Data: same as linear regression model**
* **Handling Missing Values:**

**One-Hot Encoding** is a new approach used like if its hot it cannot be cold at the same time.

**Label Encoding**

**Normalization** fixes this by putting all the numbers on the same scale.

**Min-Max Scaling** does the exact same thing, but it is only applicable to numerical columns.

**Splitting the Dataset** same as regression model. Visualization and EDA are also same.

**Model Training & Evaluation:**

**Logistic Regression**: it fits the model in training data.  
**Decision Tree**: Selected for its ability to handle non-linear relationships.   
**Random Forest**: It combines multiple decision trees to improve accuracy It is chosen for its high performance.

**Model Evaluation and Comparison:**  
The final step compares the performance of the three models.  
Performance Summary:

* Logistic Regression: 85.08%
* Decision Tree: 90.83%
* Random Forest: 92.20%

**Conclusion**:

The Random Forest model is identified as the best-performing model, achieving the highest accuracy of 92.20%.

**Linear Regression:** Predicts a number

**Classification:** Predicts a category