Car Price Prediction

LINEAR REGRESSION MODEL

Contents

Problem Statement:

The problem addressed in this case study is car price prediction.

Given customer attributes such as age, annual salary, credit card debt, and net worth, the objective here is to predict the amount a customer is likely to spend on purchasing a car.

Reason for selection:

1. It is a model for performing tasks that involve regression like price prediction.

2. It identifies relationships between input features and the target variable that are non-linear.

3. It helps predict prices that may not have a simple linear correlation with customer attributes.

4.The model can learn from data patterns and generalize it to make accurate predictions.

Possible Applications of the Model:

1. **Car Dealerships**: Car dealerships can use this model to estimate how much a customer is likely to spend on a car, enabling them to offer tailored options and promotions.

2. **Financial Institutions**: Banks and credit institutions can use this information to assess a customer's creditworthiness and make loan decisions.

3. **Marketing**: Businesses can use the model to target potential car buyers with personalized advertising and promotions.

4. **Customer Insights**: The model can provide insights into customer behavior and preferences based on their financial profiles.

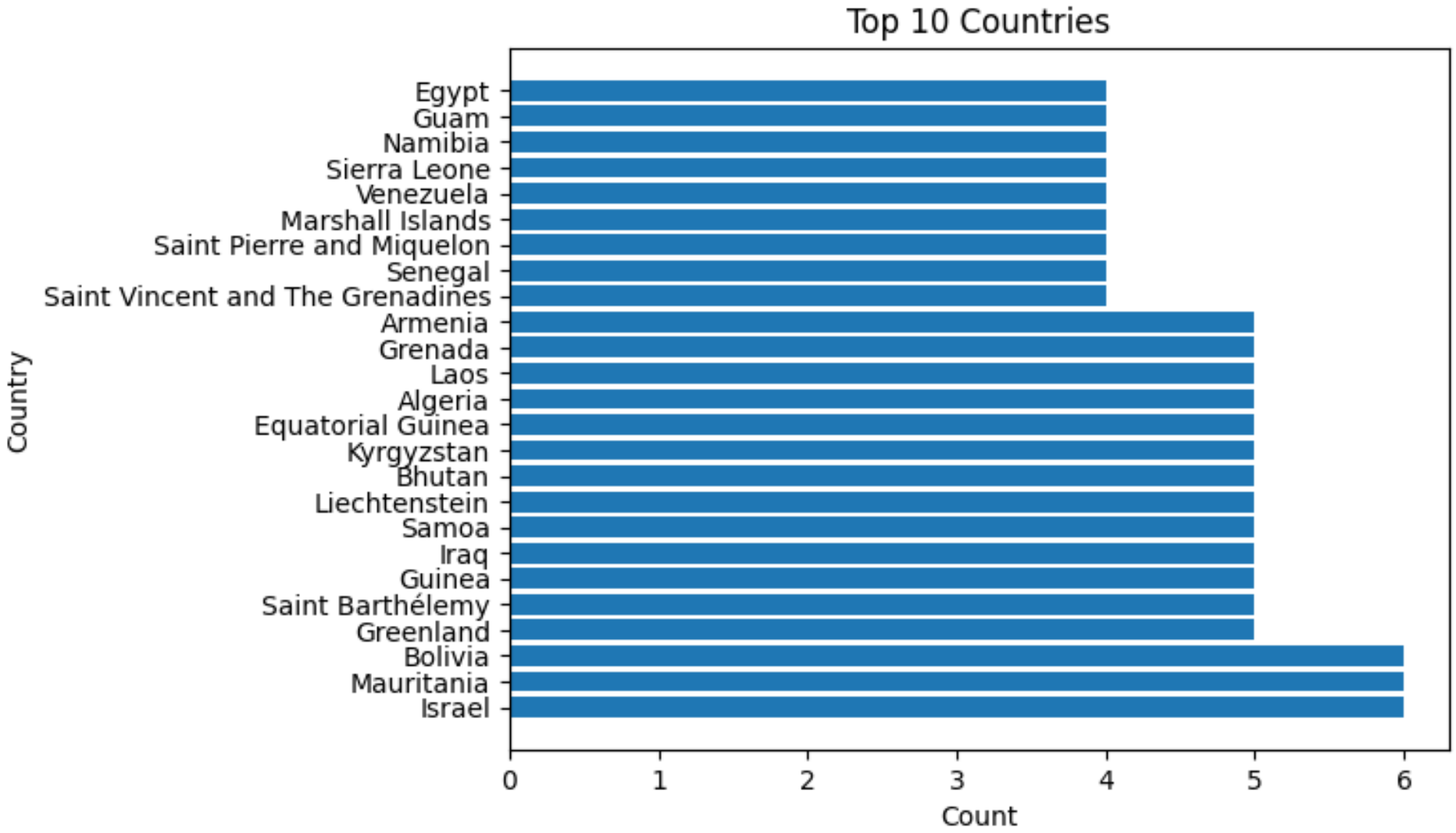
PREPROCESSING

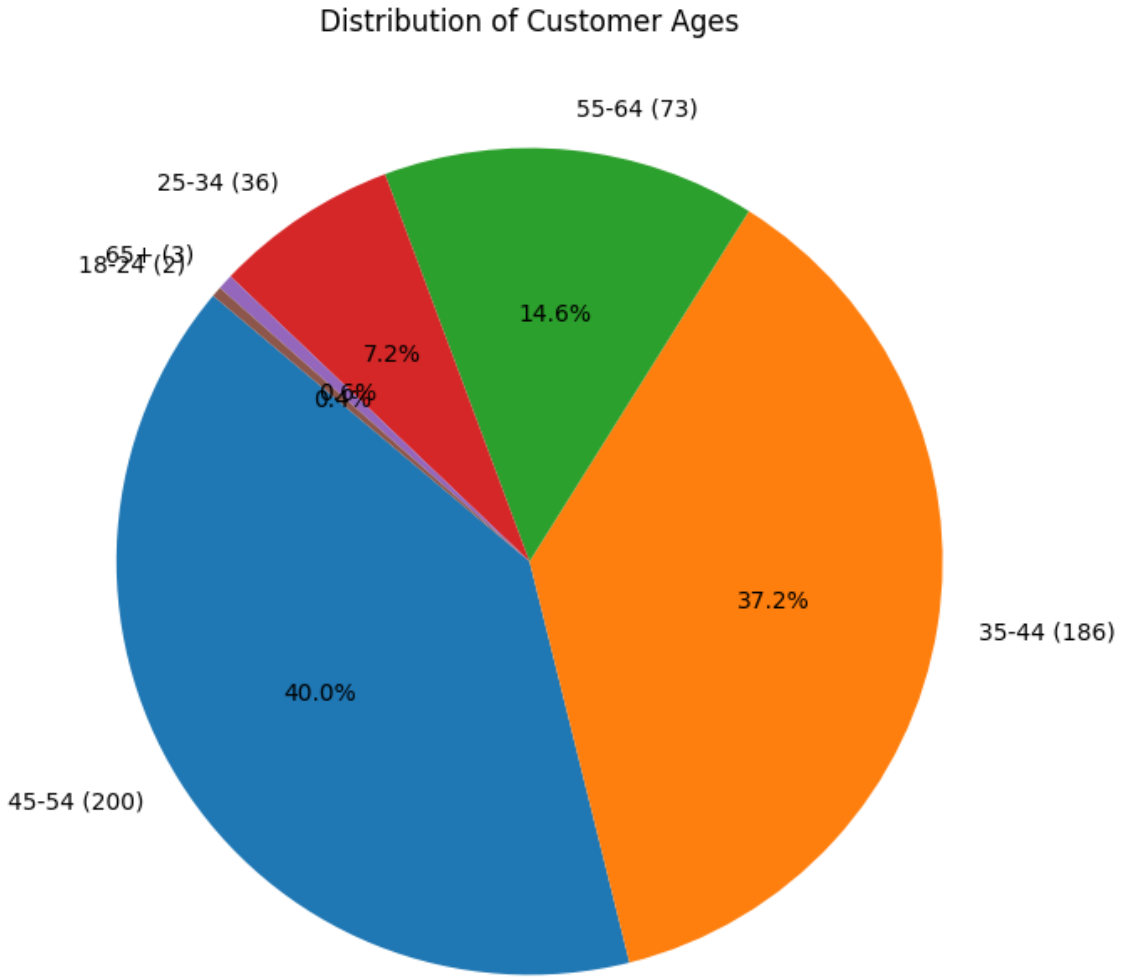
Data Preprocessing:

We imported the necessary libraries such as numpy, pandas, matplotlib, tensorflow and keras.

The dataset was then loaded using pandas, and data exploration was performed, including identifying :

1. Country with Highest Number of Car Purchases
2. Bar Graph on Top 25 Countries with the Most Customers
3. Pie Chart on Customer Ages


Top 10 Countries
Egypt
Guam
Namibia
Sierra Leone
Venezuela
Marshall Islands
Saint Pierre and Miquelon
Senegal
Saint Vincent and The Grenadines
Armenia
Grenada
Laos
Algeria
Equatorial Guinea
Kyrgyzstan
Bhutan
Liechtenstein
Samoa
Iraq
Guinea
Saint Barthélemy
Greenland
Bolivia
Mauritania
Israel
Country
0
1
2
3
4
5
6
Count


Distribution of Customer Ages
55-64 (73)
25-34 (36)
65+ (3)
18-24 (2)
14.6%
7.2%
0.6%
0.4%
37.2%
35-44 (186)
40.0%
45-54 (200)

Data Selection:

The unnecessary columns which did not contribute to the prediction were dropped, which included:

1. Customer name
2. Customer e-mail
3. Country
4. Gender

The 4 features (age, annual salary, credit card debt, net worth) were scaled using **MinMaxScaler ( )** function.

The dataset was split into training and testing sets using **train_test_split**.

'df.head()' was used to print the first 5 rows of the Dataframe

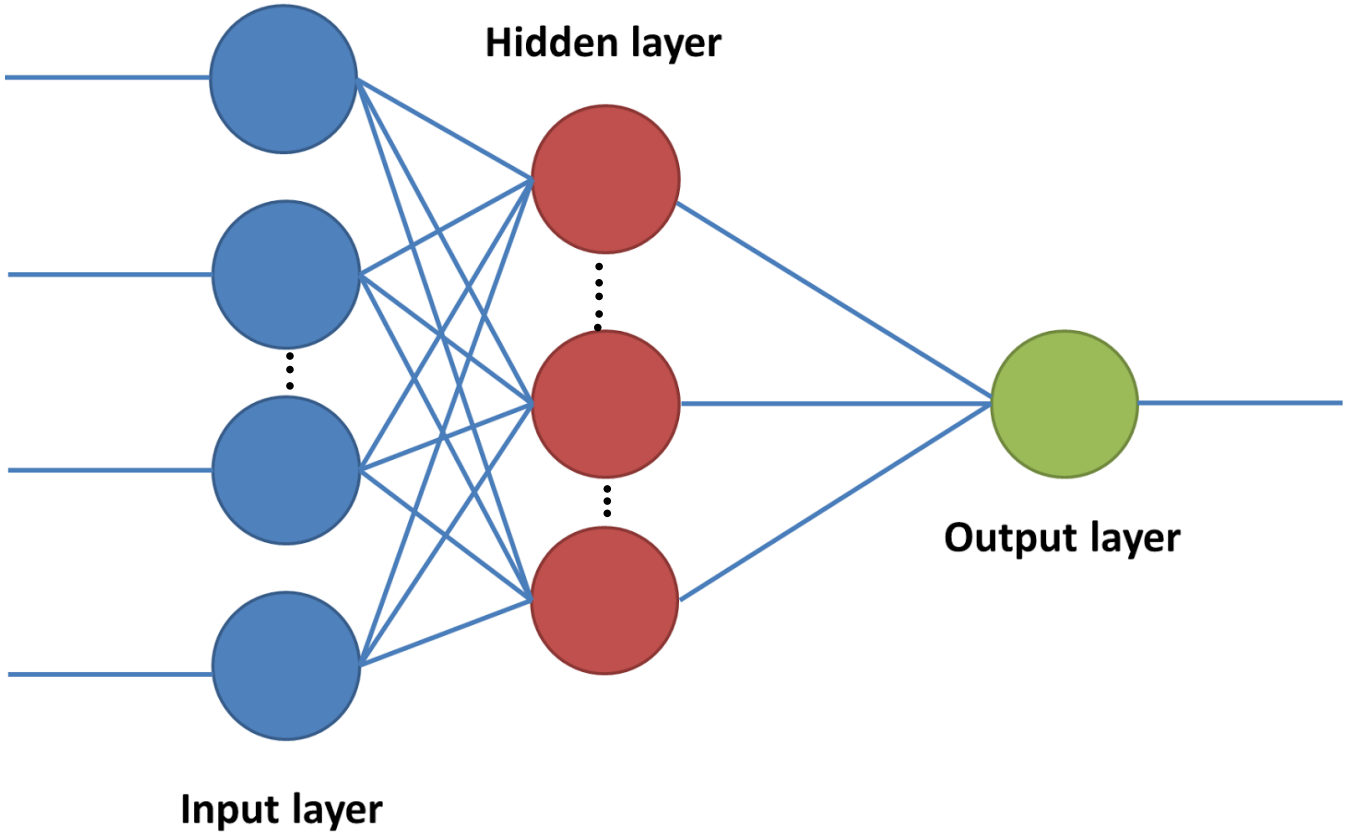
	customer name	customer e-mail	country	gender	age	annual Salary	credit card debt	net worth	car purchase amount
0	Martina Avila	cubilia.Curae.Phasellus@quisaccumsanconvallis.edu	Bulgaria	0	41.851720	62812.09301	11609.380910	238961.2505	35321.45877
1	Harlan Barnes	eu.dolor@diam.co.uk	Belize	0	40.870623	66646.89292	9572.957136	530973.9078	45115.52566
2	Naomi Rodriquez	vulputate.mauris.sagittis@ametconsectetueradip...	Algeria	1	43.152897	53798.55112	11160.355060	638467.1773	42925.70921
3	Jade Cunningham	malesuada@dignissim.com	Cook Islands	1	58.271369	79370.03798	14426.164850	548599.0524	67422.36313
4	Cedric Leach	felis.ullamcorper.viverra@egetmollislectus.net	Brazil	1	57.313749	59729.15130	5358.712177	560304.0671	55915.46248

ABOUT THE MODEL

Model architecture:

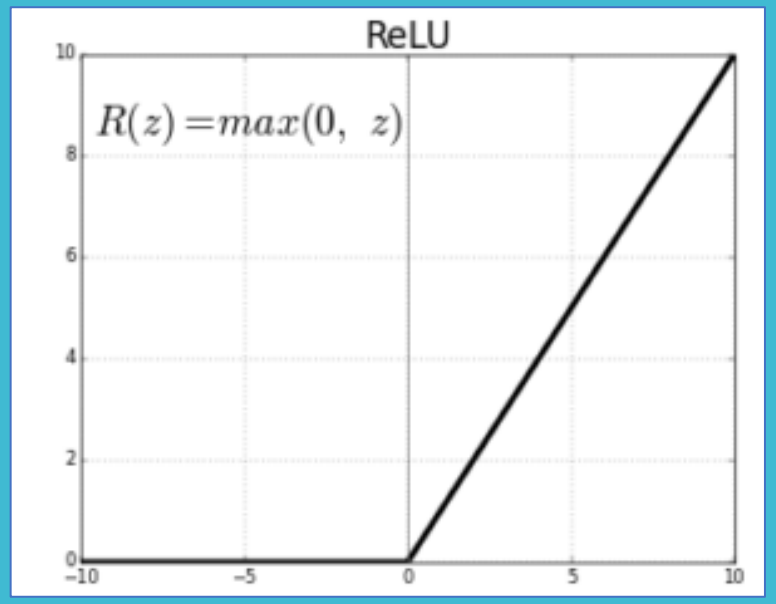
The ANN model consists of three layers:

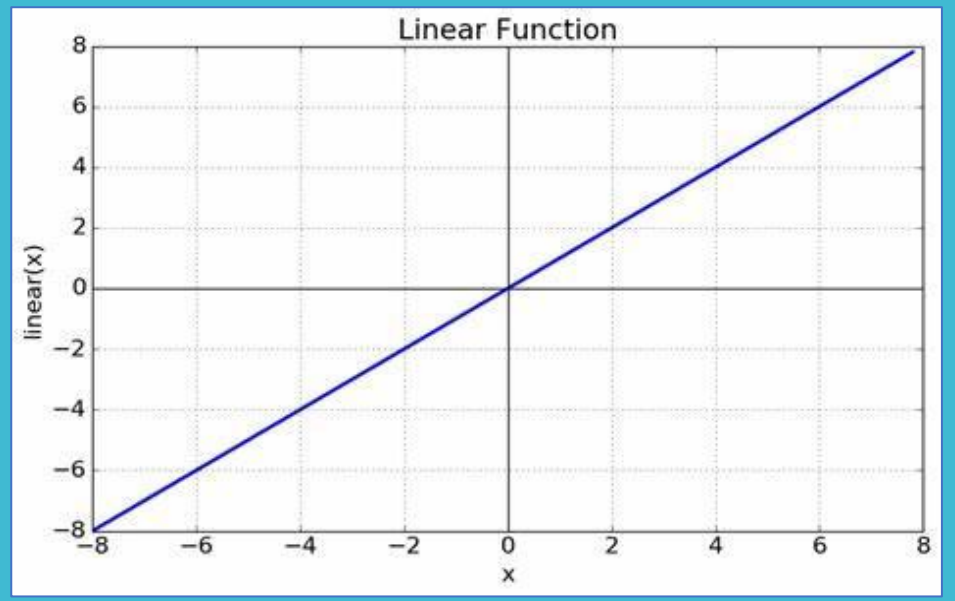
Layer 1 – 25 neurons (input layer)
Layer 2 – 10 neurons (hidden layer)
Layer 3 – 1 neurons (output layer)



Activation functions used:

1. Relu Activation function
2. Linear Activation function





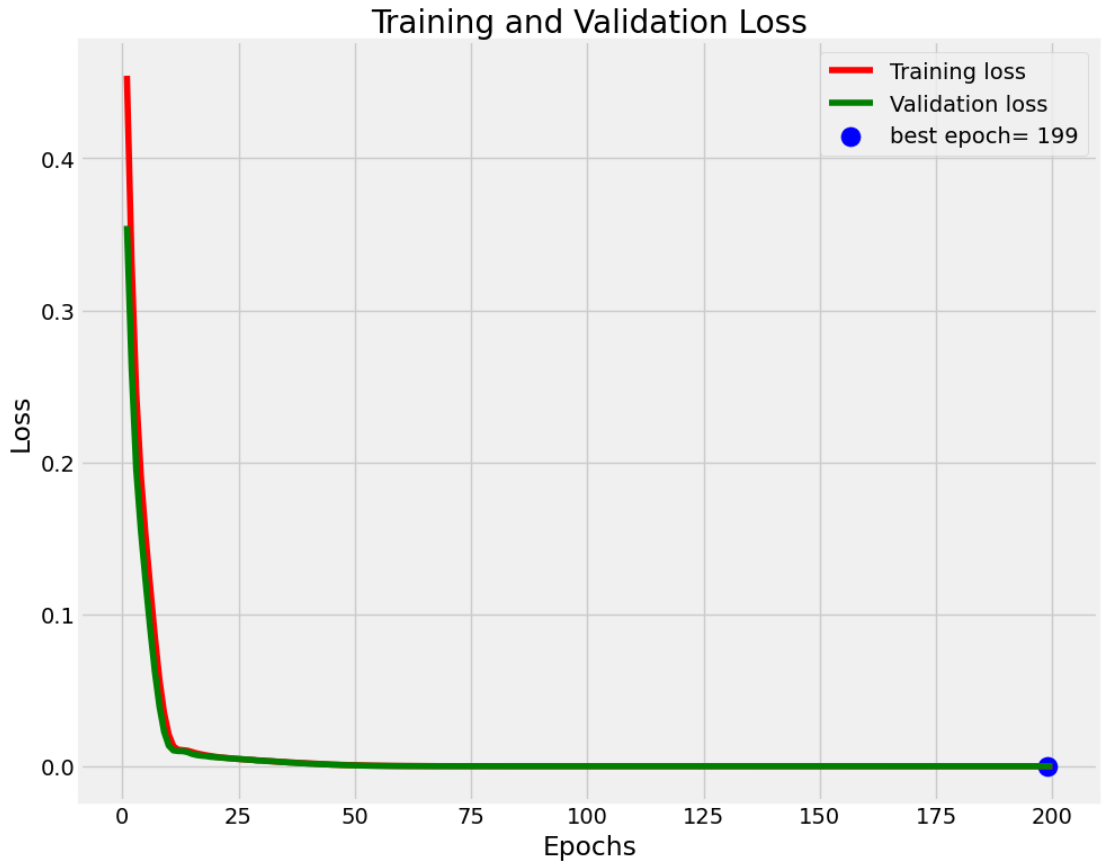
Model compilation:

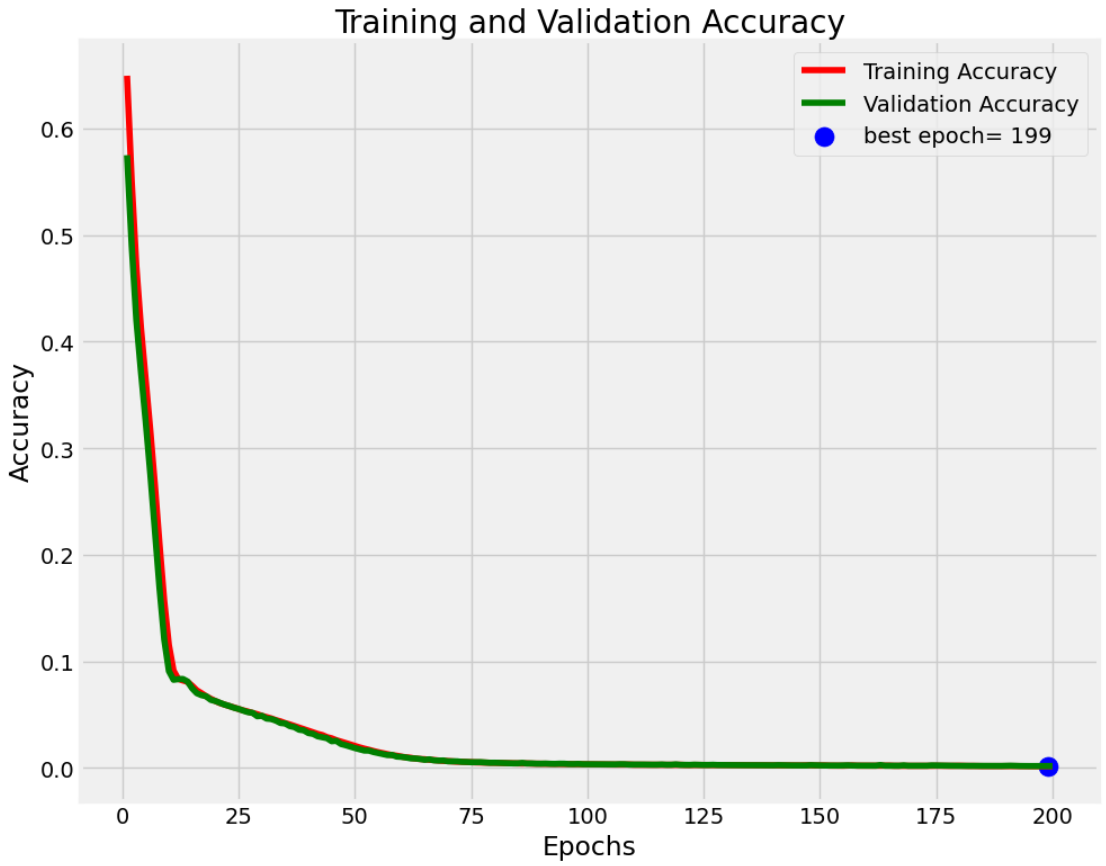
The following were used to compile the model:

Optimizer : Adam optimizer
Loss : Mean squared error
Metrics: Mean absolute error

Metrics include, Training loss, Training accuracy, Validation loss and Validation accuracy.

Using model history, Training loss, Training accuracy, Validation loss and Validation accuracy were plotted and it was observed that the best epoch is the **199**th iteration.





Car Price Prediction:

We used the trained model to predict the car price for a sample customer and calculated the mean squared error.

```
new_data = np.array([[41.851720, 62812.09301, 11609.380910, 238961.2505]])
```

```
1/1 [==============================] - 0s 20ms/step
Predicted Price: 43280.637
Actual Price: 35321.45877
mean square error: 3.167425680993415
```

The linear regression model was applied on the test data which was **25%** of the total data and a remaining training data of **75%**.

The accuracy was calculated by using the R-squared (R2) score which is,

$$R^2 = (r)^2$$

This represents how close the model is to the actual output.

Interpreting the coefficient of determination

Coefficient of determination (R^2)	Interpretation
0	The model **does not** predict the outcome.
Between 0 and 1	The model **partially** predicts the outcome.
1	The model **perfectly** predicts the outcome.

The model's R2 score was **0.9985301745950076**, which is very close to **1** and this indicates a strong ability to predict car purchase amounts.

Case Study by:

Krishna Varma Vetukuri
Hu21csen0300138

Patham Sravya
Hu21csen0300159