Index

CARBON DIOXIDE EMISSION PREDICTION USING LINEAR REGRESSION

ABSTRACT

In the current research landscape, power of data to make a sizeable contribution to prediction algorithms is huge. We can use prediction algorithms like Linear regression in business to evaluate trends and to make forecasts/estimates, for example if the sales in a company have increased exponentially for the last two years, we can conduct linear analysis on the monthly sales data and predict the sales in the future years. For our assignment we have considered a dataset which provides the model specific fuel consumption ratings and estimated carbon and the estimated for new light duty vehicles for retail sale in Canada in 2022

Using the above-mentioned dataset, we use the model to predict the CO2 emissions(tailpipe), in grams per kilometre which is a combination of two parameters(city and highway) as present in the dataset, we have picked this particular dataset as it is tackling an issue which is extremely vital for the environment and for vehicle manufactures as it highlights the carbon dioxide emission, this model can aid the vehicle manufactures in picking out certain vehicles which can cause more damage to the environment through the predicted carbon dioxide emissions for the vehicles under consideration. We have also created certain key performance indicators through which aid in comparing different aspects of the data under consideration which in turns enables the vehicle manufacturers to understand the performance of their products which would then enhance their future business decisions with respect to their products.

CHOICE OF DEPENDENT AND INDEPENDENT VARIABLES AND SELECTION OF ALGORITHM

Predictive algorithms have gained traction over the past few decades, there has been a significant amount of research conducted in this area. For our assignment we have considered 15 variables, namely:

1) MODEL_YEAR: This denotes the year under consideration for a particular vehicle

2) BRAND: This denotes the company of the vehicle

3) MODEL: This denotes the model type/range of vehicle as specified by the company

4) VEHICLE_CLASS: This denotes the segments of automotive vehicles for the purpose of vehicle emissions control and fuel economy calculation

5) ENGINE_SIZE_L: This denotes the volume of fuel and air that can be pushed through the vehicle's cylinder

6) CYLINDER: This denotes the number of cylinders present in the vehicle

7) TRANSMISSION: This denotes the type of gear transmission in the vehicle **(A = automatic; AM = automated manual; AS = automatic with select shift; AV = continuously variable; M = manual; 3 – 10 = Number of gears)**

8) FUEL TYPE: This denotes the type of fuel used by the vehicle (X = Regular gasoline; Z = Premium gasoline)

9) FUEL CONSUMPTION(CITY(L/100): This denotes the fuel consumed by the vehicle to cover 100 kms in city limits

10) FUEL CONSUMPTION(HWY(L/100): This denotes the fuel consumed by the vehicle to cover 100 kms on the highway

11) FUEL CONSUMPTION(Comb(L/100): This denotes the fuel consumed by the vehicle to cover 100 kms with a combination of the city and the highway in a combined rating (55% city, 45% highway)

12) FUEL CONSUMPTION(Comb(L/100): This denotes the fuel consumed by the vehicle to cover 100 kms with a combination of the city and the highway in a combined rating (55% city, 45% highway) but expressed in miles per gallon

13) Co2 EMISSIONS: This denotes the tailpipe emissions of carbon dioxide (in grams per kilometre) for combined city and highway driving

14) Co2 RATING: This denotes tailpipe emissions of carbon dioxide rated on a scale from 1-10, where 1 is the worst and 10 being the best

15) SMOG RATING: This denotes tailpipe emissions of smog pollutants rated on a scale from 1-10, where 1 is the worst and 10 being the best

We have considered multiple cases in our analysis in which we have considered independent and dependent variables for each individual case and we have compared which among them has performed better in terms of the performance metrics

Linear Regression has been opted for modelling in our assignment as it performs the task to predict and a dependent variable value(y) based on a given set of independent variables(x) and since we have to predict the best performing model for continuous data

Since there are multiple input variables and one output variable, linear regression is the best choice under "MUTIPLE SUPERVISED MACHINE LEARNING ALGORITHMS"

The biggest advantages of linear regression models are:

Linearity

It makes the assessment strategy basic and, in particular, these direct conditions have a straightforward understanding on a measured level [1]

Simple implementation

A very straightforward procedure, linear regression can be used to provide results that are acceptable. In addition, compared to other complicated methods, these models may be trained quickly and effectively even on systems with less CPU capability. Comparing linear regression to some of the other machine learning techniques, linear regression has a significantly lower temporal complexity. The linear regression's mathematical formulae are also quite simple to comprehend and interpret. As a result, learning linear regression is quite simple. [2]

The link from where we have accessed the dataset is from KAGGLE [3]

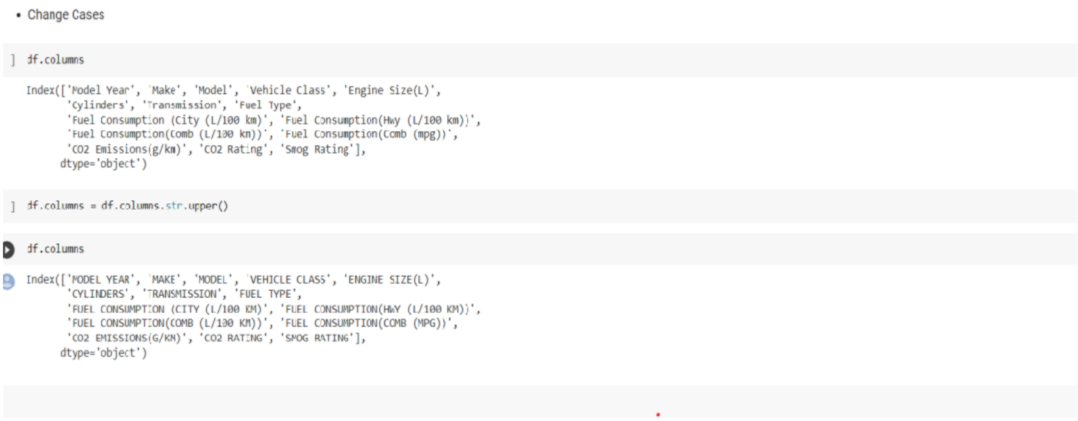
DATA PREPARATION

The process of altering raw data so that we may use machine learning algorithms to find insights or make predictions is known as "data preparation" (sometimes referred to as "data pre-processing")

Datasets typically need considerable preparation before they can produce significant insights because the majority of machine learning algorithms require data to be structured in a fairly specific way. Some datasets contain missing, incorrect, or otherwise challenging-for-an-algorithm-to-process values. The algorithm cannot use missing data. Invalid data causes the algorithm to provide less accurate or even false results. Although some datasets need to be shaped, they are generally clean [4]

We have adopted the following steps to prepare the data

1) Convert the names of columns of data into upper case so that it's more readable for analysing the data



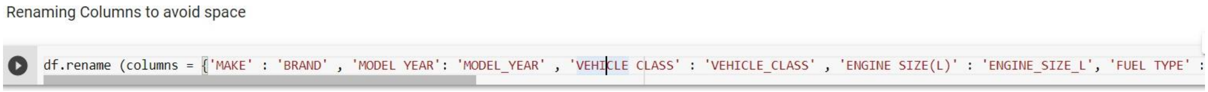
- Change Cases

```
] df.columns

Index(['Model Year', 'Make', 'Model', 'Vehicle Class', 'Engine Size(L)',
       'Cylinders', 'Transmission', 'Fuel Type',
       'Fuel Consumption (City (L/100 km)', 'Fuel Consumption(Hwy (L/100 km))',
       'Fuel Consumption(Comb (L/100 km))', 'Fuel Consumption(Comb (mpg))',
       'CO2 Emissions(g/km)', 'CO2 Rating', 'Smog Rating'],
      dtype='object')

] df.columns = df.columns.str.upper()

df.columns

Index(['MODEL YEAR', 'MAKE', 'MODEL', 'VEHICLE CLASS', 'ENGINE SIZE(L)',
       'CYLINDERS', 'TRANSMISSION', 'FUEL TYPE',
       'FUEL CONSUMPTION (CITY (L/100 KM)', 'FUEL CONSUMPTION(HWY (L/100 KM))',
       'FUEL CONSUMPTION(COMB (L/100 KM))', 'FUEL CONSUMPTION(COMB (MPG))',
       'CO2 EMISSIONS(G/KM)', 'CO2 RATING', 'SMOG RATING'],
      dtype='object')
```

Fig 1.1

2)

Renaming a few columns so that it can help us gauge the data better in terms of the familiarity i.e. (Make -> Brand) Brand here represents the name of the vehicle company under consideration. Underscore ("_") has been added to all the column names which had two words in the name, so that there is a sense of uniformity w.r.t to the column names i.e. (VEHICLE CLASS -> VEHICLE_CLASS). Also, all the parenthesis has been removed from the column names to optimize the column name i.e. (ENGINE SIZE(L)': 'ENGINE_SIZE_L)



Renaming Columns to avoid space

```
df.rename (columns = {'MAKE' : 'BRAND' , 'MODEL YEAR': 'MODEL_YEAR' , 'VEHICLE CLASS' : 'VEHICLE_CLASS' , 'ENGINE SIZE(L)' : 'ENGINE_SIZE_L', 'FUEL TYPE' :
```

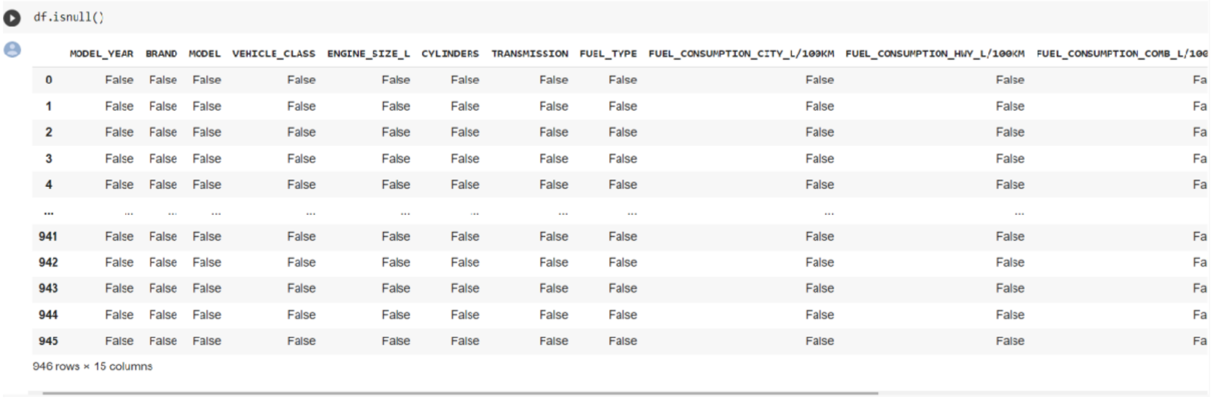
Fig 1.2

3) We have deployed the isnull() function to check for the columns for missing values

The output object will contain Boolean True/False values that indicate which values are missing.

Values that count as "missing" are None and numpy,NaN. Values like an empty string (i.e., '') or numpy.inf will not count as missing values when you use the isnull() method.[5]

If the value comes out to be False, this indicates that it does not have any missing values



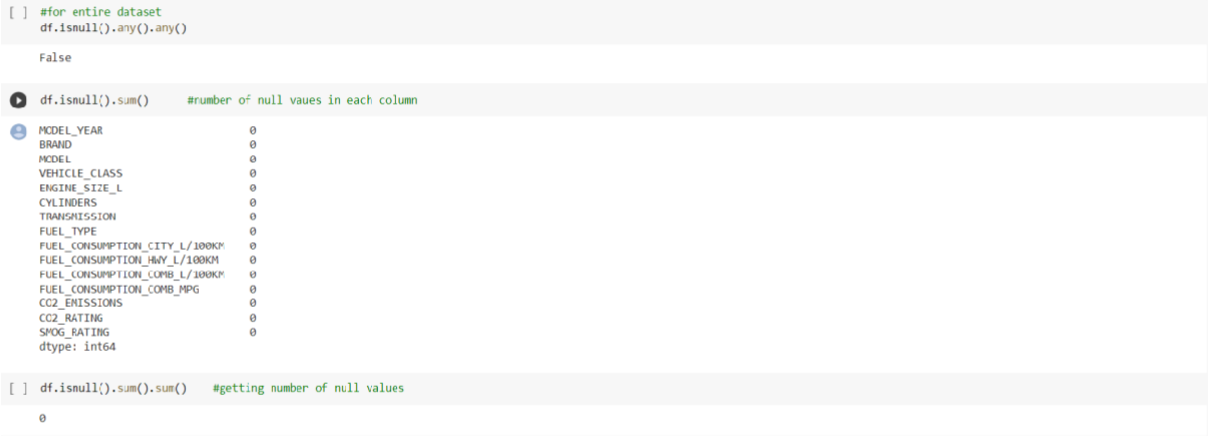
```
df.isnull()
```

	MODEL_YEAR	BRAND	MODEL	VEHICLE_CLASS	ENGINE_SIZE_L	CYLINDERS	TRANSMISSION	FUEL_TYPE	FUEL_CONSUMPTION_CITY_L/100KM	FUEL_CONSUMPTION_HWY_L/100KM	FUEL_CONSUMPTION_COMB_L/100
0	False	False	False	False	False	False	False	False	False	False	Fa
1	False	False	False	False	False	False	False	False	False	False	Fa
2	False	False	False	False	False	False	False	False	False	False	Fa
3	False	False	False	False	False	False	False	False	False	False	Fa
4	False	False	False	False	False	False	False	False	False	False	Fa
...	...	...	...	...	...	...	...	...	...	...	
941	False	False	False	False	False	False	False	False	False	False	Fa
942	False	False	False	False	False	False	False	False	False	False	Fa
943	False	False	False	False	False	False	False	False	False	False	Fa
944	False	False	False	False	False	False	False	False	False	False	Fa
945	False	False	False	False	False	False	False	False	False	False	Fa

946 rows × 15 columns

Fig 1.3

The .any() method is used to check whether there are any missing values in the particular column/dataset under consideration .It is depicted in the image that there are no missing values present in the data present in the dataset under consideration



```
[ ] #for entire dataset
    df.isnull().any().any()

    False
```

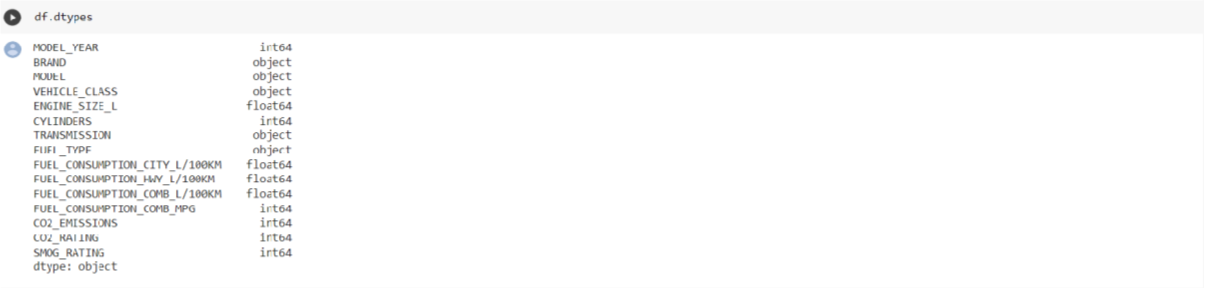
```
df.isnull().sum()       #number of null vaues in each column

MODEL_YEAR                      0
BRAND                           0
MODEL                           0
VEHICLE_CLASS                   0
ENGINE_SIZE_L                   0
CYLINDERS                       0
TRANSMISSION                    0
FUEL_TYPE                       0
FUEL_CONSUMPTION_CITY_L/100KM   0
FUEL_CONSUMPTION_HWY_L/100KM    0
FUEL_CONSUMPTION_COMB_L/100KM   0
FUEL_CONSUMPTION_COMB_MPG       0
CO2_EMISSIONS                   0
CO2_RATING                      0
SMOG_RATING                     0
dtype: int64
```

```
[ ] df.isnull().sum().sum()    #getting number of null values

    0
```

Fig 1.4

The .sum() function indicates the total sum of the null values present in the (Dataframe or series).It is specified in the image above that there are no null values present in the dataset.

4) Check for the different types present in the data columns



```
df.dtypes
```

```
MODEL_YEAR                        int64
BRAND                            object
MODEL                            object
VEHICLE_CLASS                    object
ENGINE_SIZE_L                   float64
CYLINDERS                         int64
TRANSMISSION                     object
FUEL_TYPE                        object
FUEL_CONSUMPTION_CITY_L/100KM   float64
FUEL_CONSUMPTION_HWY_L/100KM    float64
FUEL_CONSUMPTION_COMB_L/100KM   float64
FUEL_CONSUMPTION_COMB_MPG         int64
CO2_EMISSIONS                     int64
CO2_RATING                        int64
SMOG_RATING                       int64
dtype: object
```

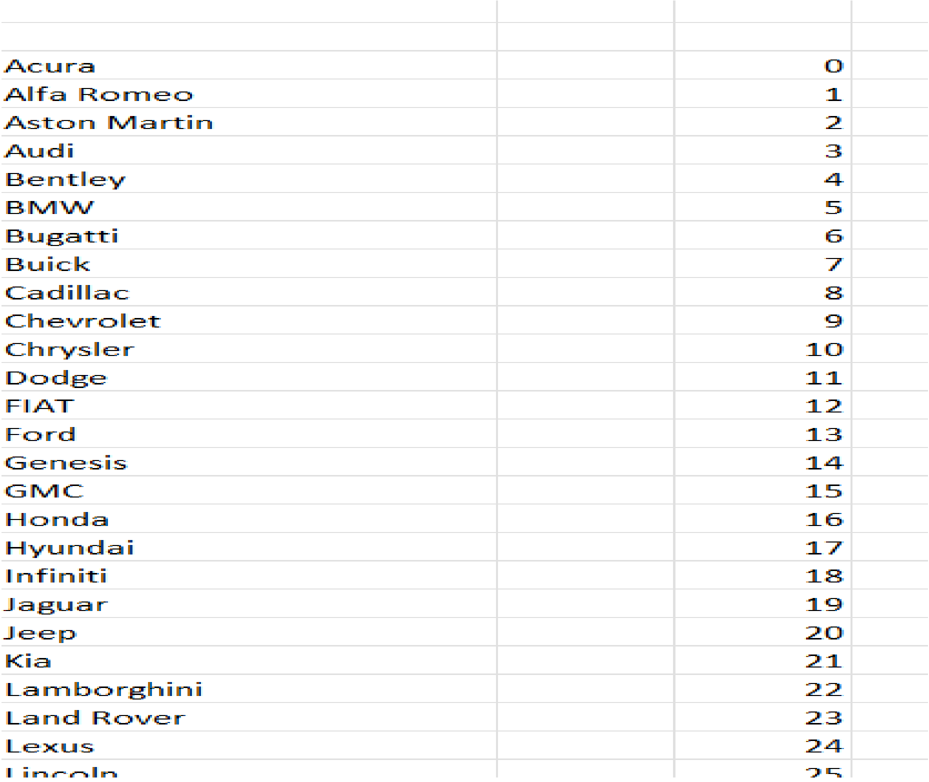
Fig 1.5

Since there are 5 data columns which are objects, we need to convert the categorical features into numerical features in this dataset, we have brands, models, vehicles class, transmission and fuel_type, so we want to do data grouping tasks on that dataframe, mathematically, we can't do any computations to the string 'Acura' or 'Alfa Romeo'. But what we can do is map a value for each of those brands that allow machine learning algorithms to do their thing:

For example: 'Acura' = 0, 'Alfa Romeo' = 1, 'Aston Martin' = 2, etc. We could do that manually using a dictionary, and just apply that mapping to a column, or we can use the le.fit() to do that for us.

So, we use it on our training data, and it will figure out the unique values and assign a value to it:

The below screengrab from an excel file just shows how the value has been mapped



Acura	0
Alfa Romeo	1
Aston Martin	2
Audi	3
Bentley	4
BMW	5
Bugatti	6
Buick	7
Cadillac	8
Chevrolet	9
Chrysler	10
Dodge	11
FIAT	12
Ford	13
Genesis	14
GMC	15
Honda	16
Hyundai	17
Infiniti	18
Jaguar	19
Jeep	20
Kia	21
Lamborghini	22
Land Rover	23
Lexus	24
Lincoln	25

Fig 1.6

DATA VISUALIZATION

We have created these graphs to aid the critical approach considered to understand the dataset

1) Brand vs Co2 Emissions

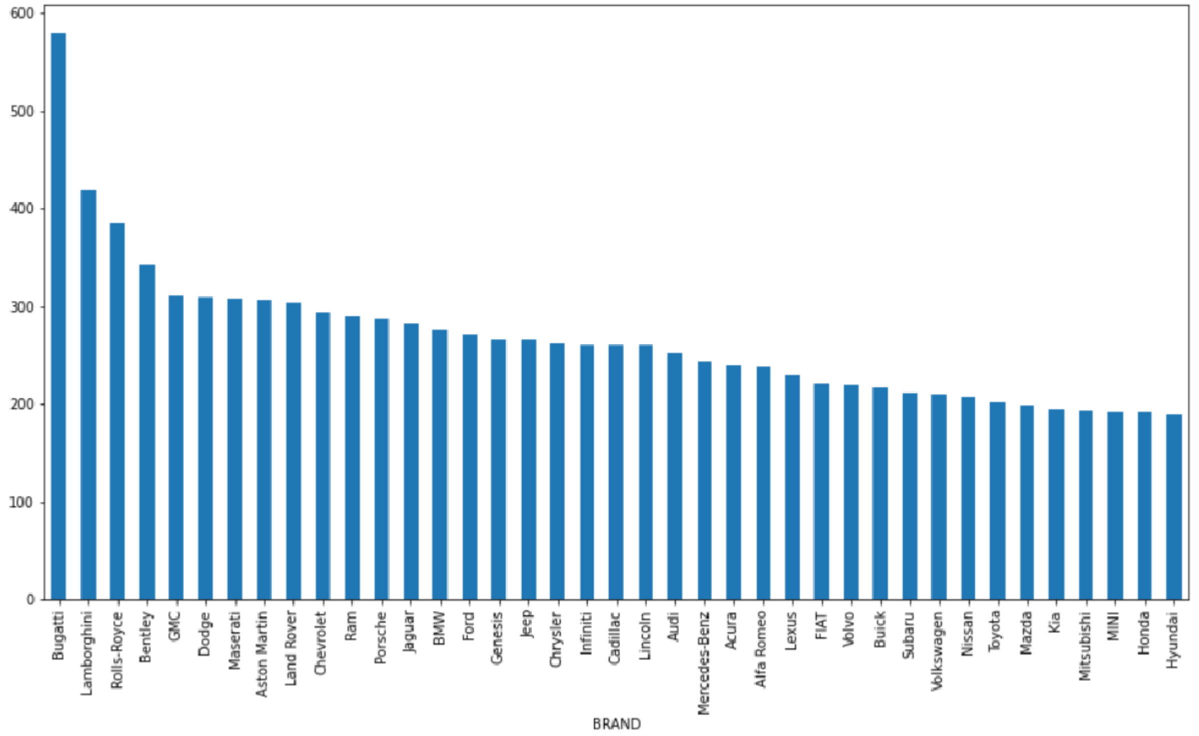


Fig 2.1

Brand on the x axis represents the names of all the vehicles present in the dataset.Co2 emissions is on the y axis which represents the Co2 emissions scale. The values in the graph are the mean values for each vehicular brand. The Highest Co2 emissions by a vehicle brand is by **BUGATTI**. The Lowest Co2 emissions by a vehicle brand is by **HYUNDAI**

2) Brand x Co2 Rating

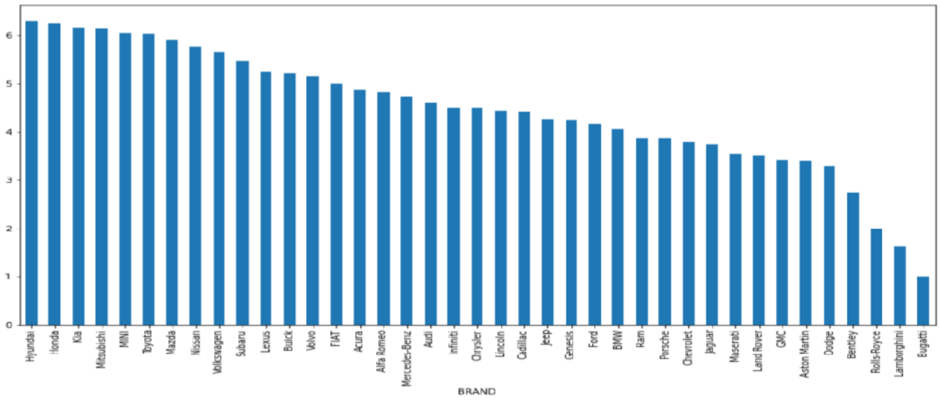


Fig 2.2

Brand on the x axis represents the names of all the vehicles present in the dataset.Co2 rating is on the y axis which represents the Co2 emission rating scale (1 (worst) to 10 (best)). The values in the graph are the mean values for each vehicular brand
The Highest Co2 emission rating by a vehicle brand is by **BUGATTI** (Worst in terms of Co2 emission)
The Lowest Co2 emission rating by a vehicle brand is by **HYUNDAI** (Best in terms of Co2 emission)

3) These set of graphs indicate the linear relationship between CO2 Emissions and Fuel Consumption_Comb_L/100 km in terms of a particular vehicular class

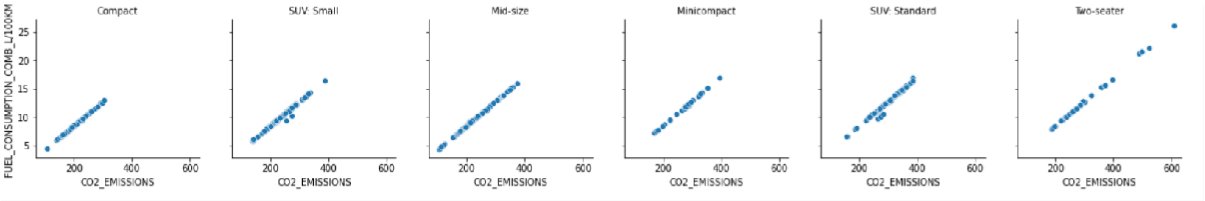


Fig 2.3

If we are to consider the first graph in the image, which for the vehicular class "COMPACT", the linear relationship indicates the with the increase in fuel consumption, there is bound to be a directly proportional increase in Co2 emissions

4) In the graph given below, the linear relationship between CO2 Emissions and Fuel Consumption_Comb_L/100 km in terms of the year under consideration, which is 2022

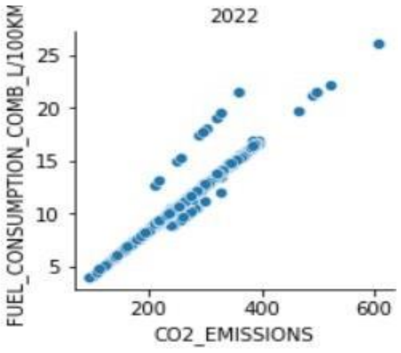


Fig 2.4

It is evident from this graph that the linear relationship between CO2 emissions and fuel consumption is prevalent throughout

5) The most significant probability distribution in statistics for independent, random variables is the normal distribution, the normal distribution defines how a variable's values are distributed, just like any probability distribution does.[6]

As you can see, the distribution of Co2 emissions follows the typical bell curve pattern for all normal distributions. Most Co2 emissions are close to the average (259.17g/km). Small differences between Co2 emissions and the mean occur more frequently than substantial deviations from the mean. The standard deviation is (64.409), which indicates the typical distance that individual vehicular Co2 emissions tend to fall from the mean.

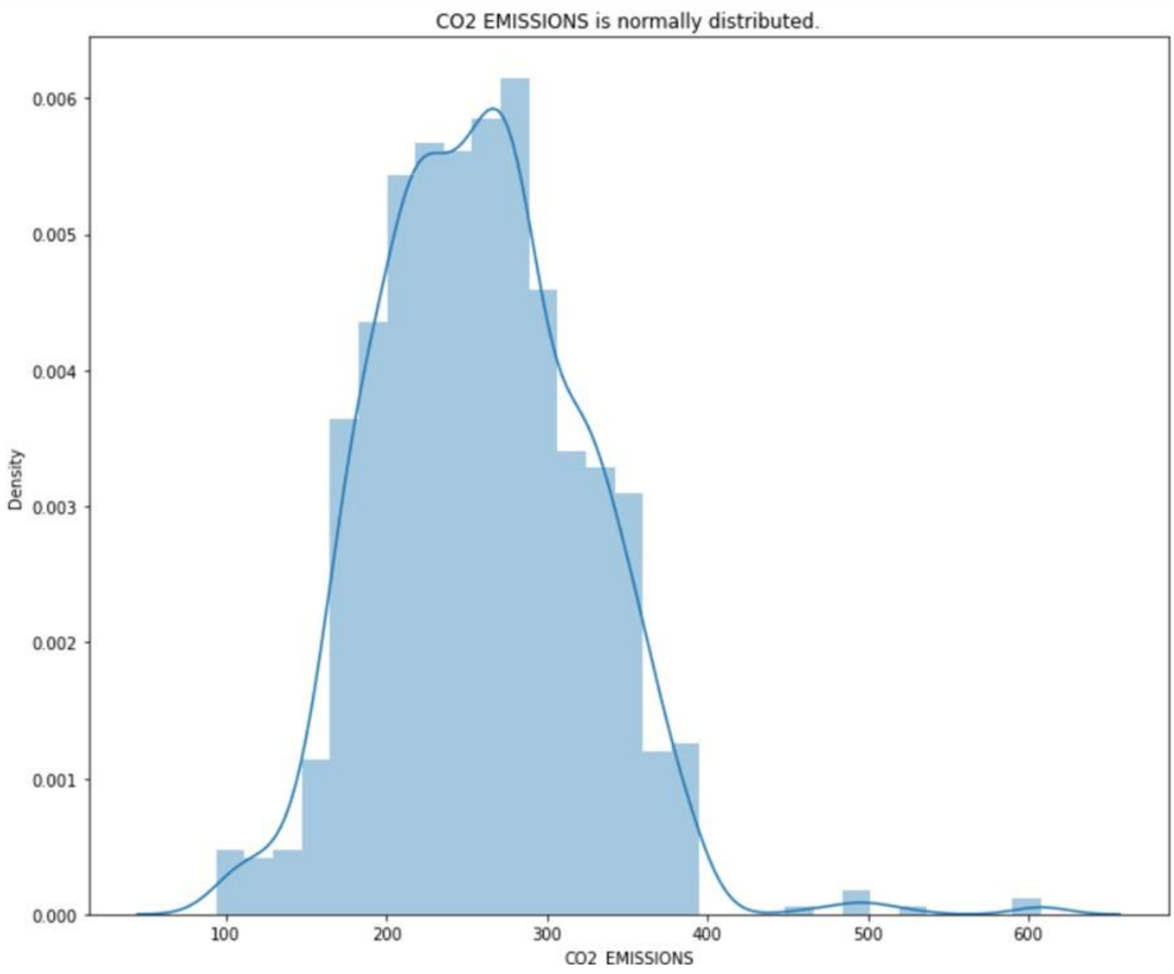


Fig 2.5

6) Box plots are used to display the distributions of numerical data values, particularly when comparing them across various groups. They are designed to give high-level information from the data under consideration, in the graph below it is used to give us Min, max, q1, q3, median, upper fence and outliers with regards to CO2 emissions.

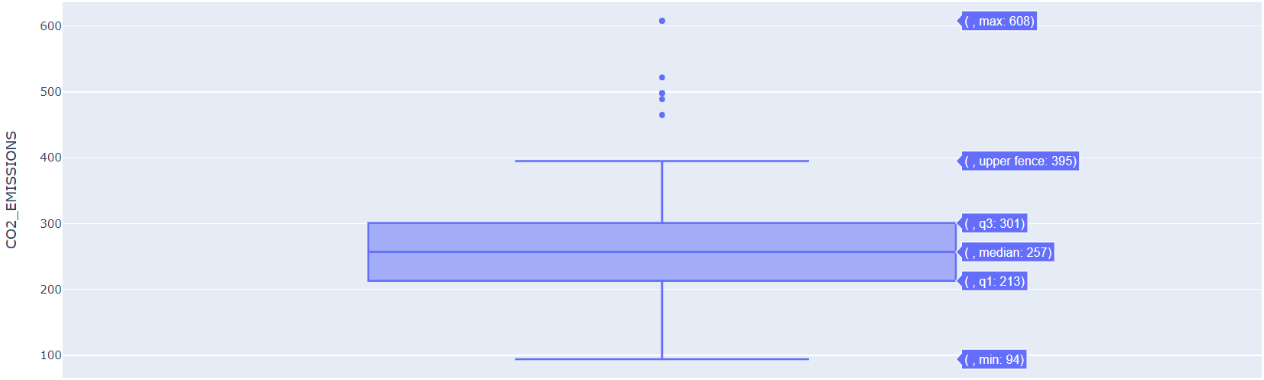


Fig 2.6

7) We have considered joint plot to establish the relationship between CO2 Emissions and Engine size in litres

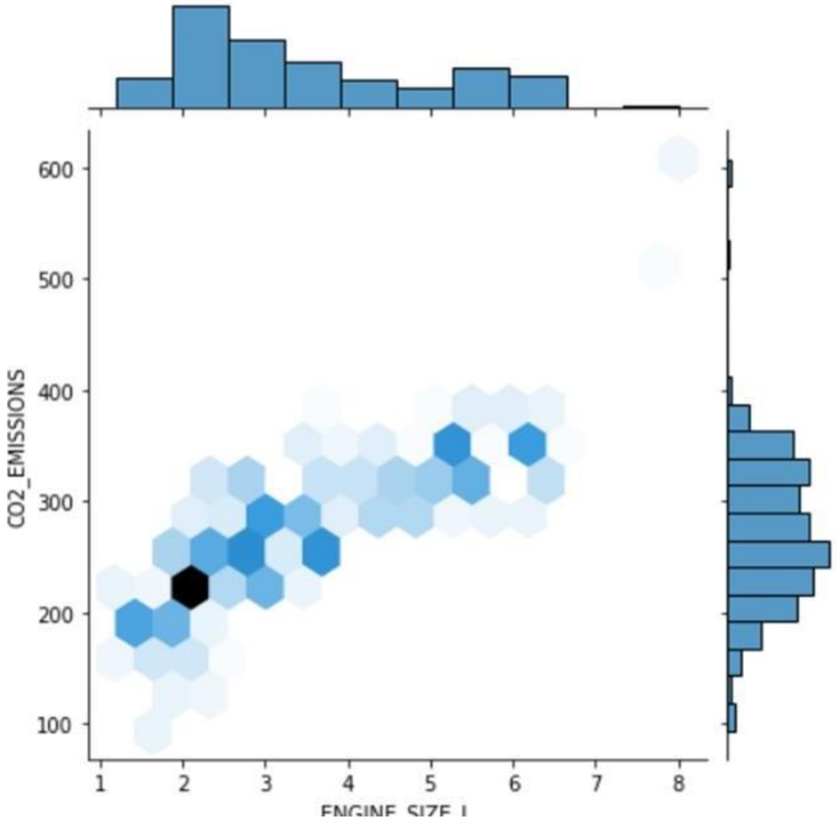


Fig 2.7

Scatter plot represents the data point for every single vehicular brand as per the dataset, we can observe that as the engine size increases there is there is an increase in the CO2 emissions and vice versa.

The bar plots in the x axis represent the marginal distribution of **Engine Size L**

The bar plots in the y axis represent the marginal distributions of **Co2 emissions**

8) We have considered a set of graphs to establish the correlation between CO2 EMISSIONS and FUEL_CONSUMPTION_COMB_L/100KM with respect to their Engine Size The graphs aid in establishing the relationship between CO2 EMISSIONS and FUEL_CONSUMPTION_COMB_L/100KM for each Individual engine size entity

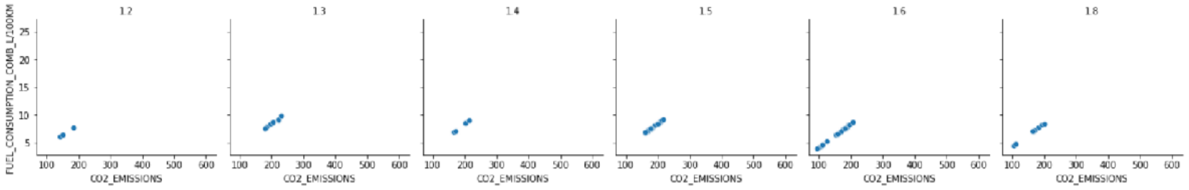


Fig 2.8

From these set of graphs, we can figure out which one among them is the best Engine Size for low CO2 Emissions.

9) This graph denotes the correlation between Fuel consumption for highway and the combination for fuel consumption (Highway + City)

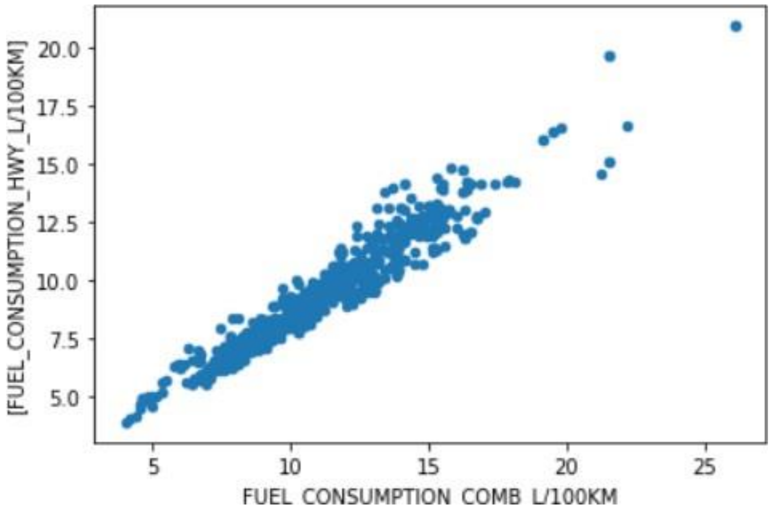


Fig 2.9

The data points present in the graph represents the fact as the fuel consumption increases for highway it results in increasing the fuel consumption for combination of highway plus city in totality.

10) This graph denotes the corelation between Fuel consumption for city and the combination for fuel consumption (Highway + City)

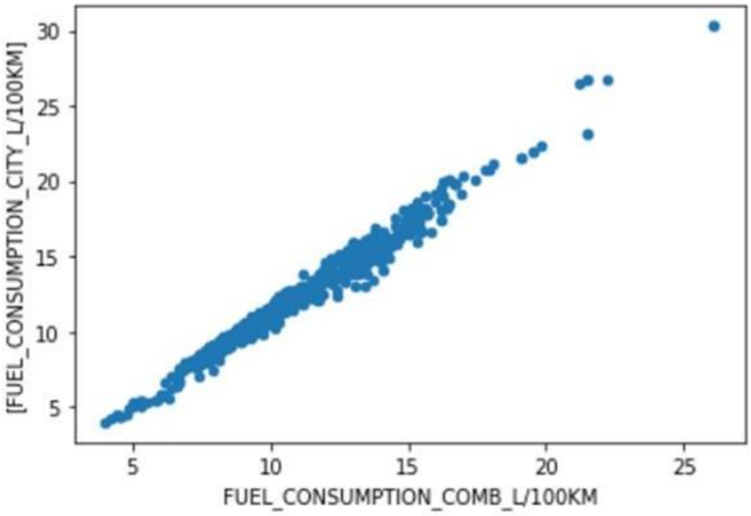


Fig 2.10

The data points present in the graph represents the fact as the fuel consumption increases for city it results in increasing the fuel consumption for combination of highway plus city in totality.

FEATURE SELECTION WITH MODEL DEVELOPMENT AND IT'S EVALUATION

As we have opted for Linear Regression and we also have picked the Stochastic Gradient descent algorithm predict which model is more helpful to provide the most accurate results and which has the most real-world implications w.r.t to the dataset considered.

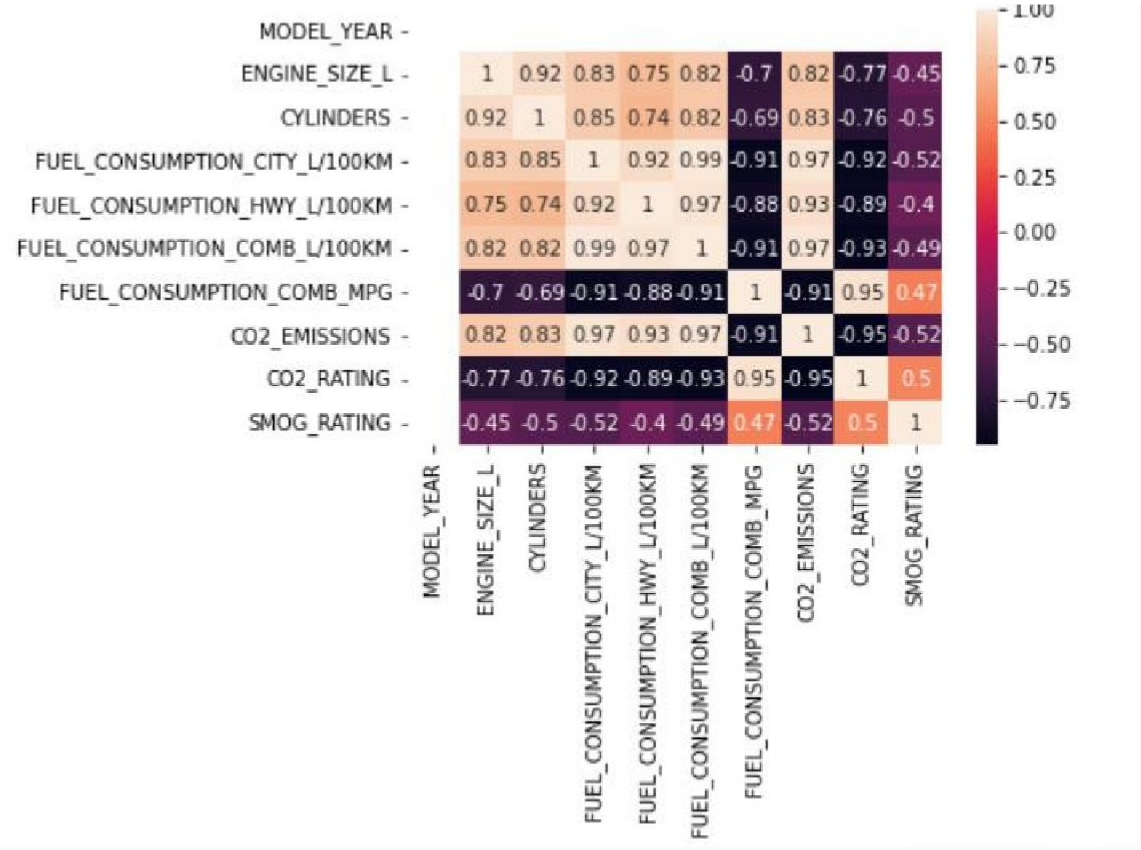


Fig 3.1

The correlation heatmap in the above image was constructed by us to understand the different independent and dependent variables which can be considered to give us the most precise model.

We have adopted three cases to make our case, all the three cases are derivatives from the correlation heatmap

This correlation heatmap consists of all the numerical features present in the dataset

We have considered three cases in our attempt to pick the best performing model

Case 1:

Independent variables:

'MODEL_YEAR','BRAND','MODEL',VEHICLE_CLASS','ENGINE_SIZE_L','FUEL_CONSUMPTION_COMB_L/100KM','TRANSMISSION','FUEL_TYPE','CO2_RATING','SMOG_RATING'

Dependent variable:

'CO2_EMISSIONS'

In our correlation matrix, the range for high correlation is 0.90-0.99

In the first case we have only considered the above mentioned independent and dependent variables because "FUEL_CONSUMPTION_CITY_L/100KM" and "FUEL_CONSUMPTION_COMB_L/100KM" have the same correlation values, this is why we opted for two cases, with Case 1 being the one without

CYLINDERS,

FUEL_CONSUMPTION_CITY_L/100KM,

FUEL_CONSUMPTION_HWY_L/100KM,

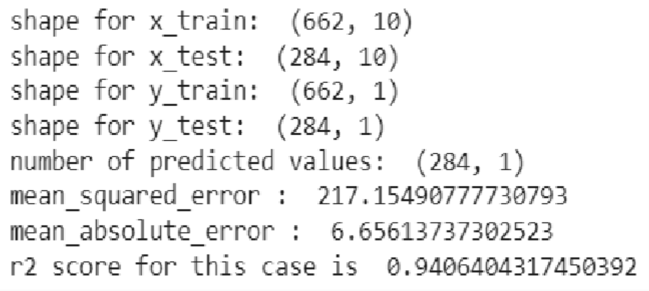
FUEL_CONSUMPTION_COMB_MPG

We have considered Stochastic Gradient Descent optimisation algorithm because

Due to the network only processing one training sample, it is simpler to fit into memory and just one sample is processed at a time, it is computationally quick.

In the implementation of Stochastic Gradient Descent algorithm in Case 1, we have considered the penalty= None, since predictions is equal to test data.

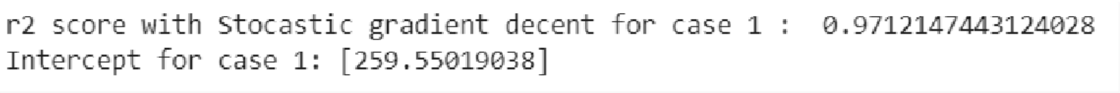
The result of the following study of linear regression with and without Stochastic Gradient Descent algorithm for Case 1 is:



```
shape for x_train:  (662, 10)
shape for x_test:  (284, 10)
shape for y_train:  (662, 1)
shape for y_test:  (284, 1)
number of predicted values:  (284, 1)
mean_squared_error :  217.15490777730793
mean_absolute_error :  6.65613737302523
r2 score for this case is  0.9406404317450392
```

Fig 3.2

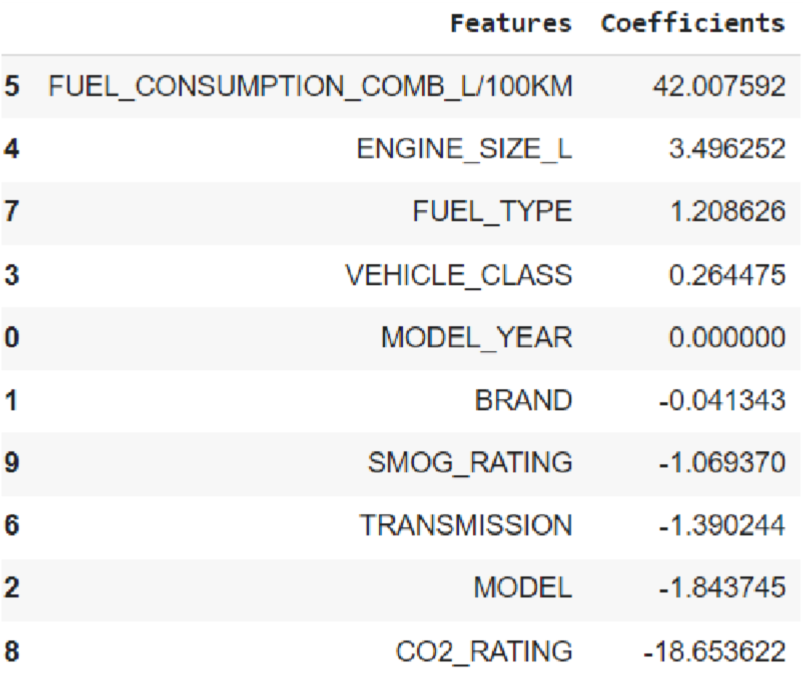
Linear Regression without Stochastic Gradient Descent algorithm



```
r2 score with Stocastic gradient decent for case 1 :  0.9712147443124028
Intercept for case 1: [259.55019038]
```

Linear Regression with Stochastic Gradient Descent algorithm

Features of the Stochastic Gradient Descent algorithm along with their coefficients are represented in this image



	Features	Coefficients
5	FUEL_CONSUMPTION_COMB_L/100KM	42.007592
4	ENGINE_SIZE_L	3.496252
7	FUEL_TYPE	1.208626
3	VEHICLE_CLASS	0.264475
0	MODEL_YEAR	0.000000
1	BRAND	-0.041343
9	SMOG_RATING	-1.069370
6	TRANSMISSION	-1.390244
2	MODEL	-1.843745
8	CO2_RATING	-18.653622

Fig 3.3

In the form of a graphical representation

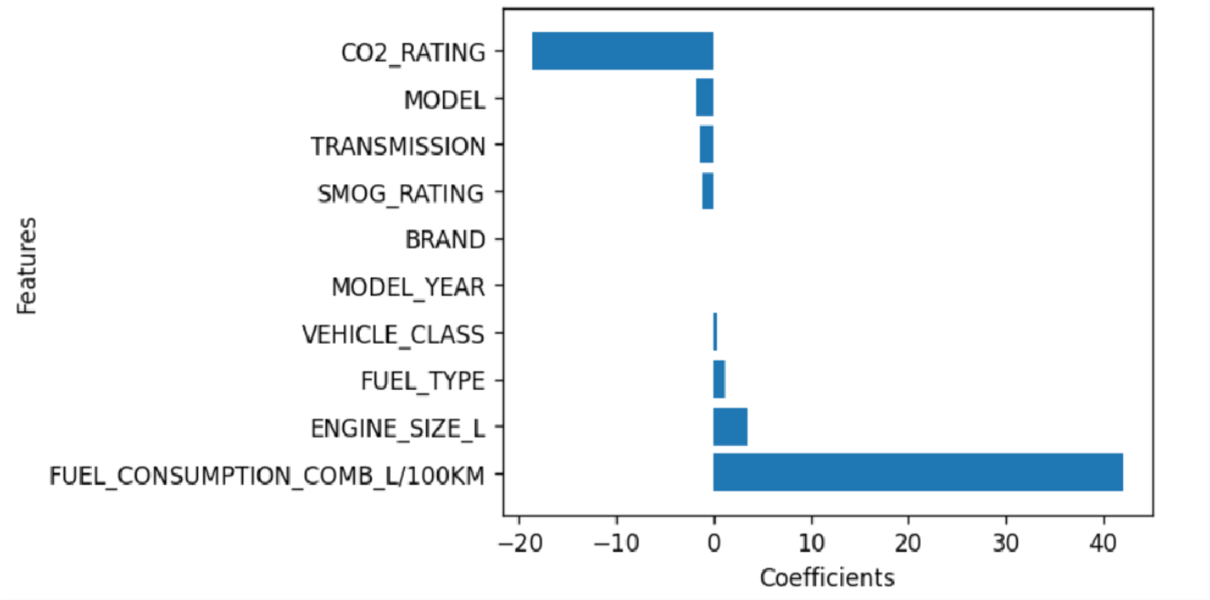


Fig 3.4

Case 2:

Independent variables:

MODEL_YEAR,

ENGINE_SIZE_L,

FUEL_CONSUMPTION_CITY_L/100KM,

FUEL_CONSUMPTION_HWY_L/100KM,

TRANSMISSION,

FUEL_TYPE,

CO2_RATING,

SMOG_RATING

Dependent variable:

CO2_EMISSIONS

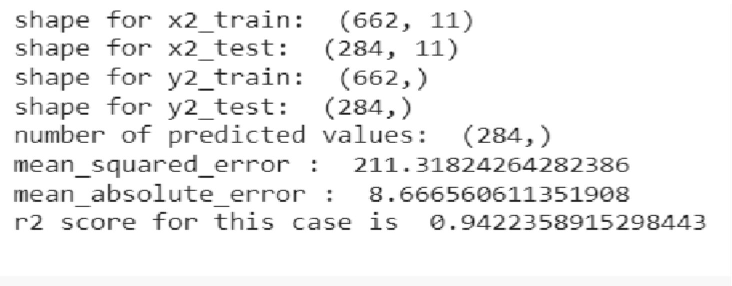
In our correlation matrix, the range for high correlation is 0.90-0.99

In the second case we have only considered the above mentioned independent and dependent variables because "FUEL_CONSUMPTION_CITY_L/100KM" and "FUEL_CONSUMPTION_COMB_L/100KM" have the same correlation values,

We have considered Stochastic Gradient Descent optimisation algorithm for this case as well

In the implementation of Stochastic Gradient Descent algorithm in Case 2, we have considered the penalty= None, since predictions is equal to test data

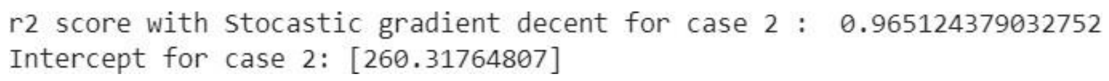
The result of the following study of linear regression with and without Stochastic Gradient Descent algorithm for Case 2 is:



```
shape for x2_train:  (662, 11)
shape for x2_test:  (284, 11)
shape for y2_train:  (662,)
shape for y2_test:  (284,)
number of predicted values:  (284,)
mean_squared_error :  211.31824264282386
mean_absolute_error :  8.666560611351908
r2 score for this case is  0.9422358915298443
```

Fig 3.5

Linear Regression without Stochastic Gradient Descent algorithm

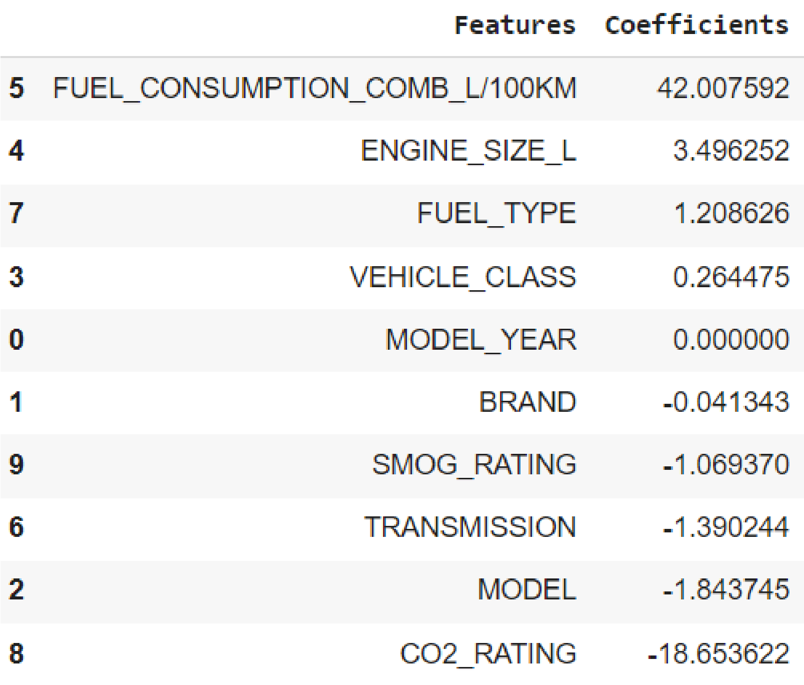


```
r2 score with Stocastic gradient decent for case 2 :  0.965124379032752
Intercept for case 2: [260.31764807]
```

Fig3.6

Linear Regression with Stochastic Gradient Descent algorithm

Features of the Stochastic Gradient Descent algorithm along with their coefficients are represented in this image



	Features	Coefficients
5	FUEL_CONSUMPTION_COMB_L/100KM	42.007592
4	ENGINE_SIZE_L	3.496252
7	FUEL_TYPE	1.208626
3	VEHICLE_CLASS	0.264475
0	MODEL_YEAR	0.000000
1	BRAND	-0.041343
9	SMOG_RATING	-1.069370
6	TRANSMISSION	-1.390244
2	MODEL	-1.843745
8	CO2_RATING	-18.653622

Fig 3.7

In the form of a graphical representation

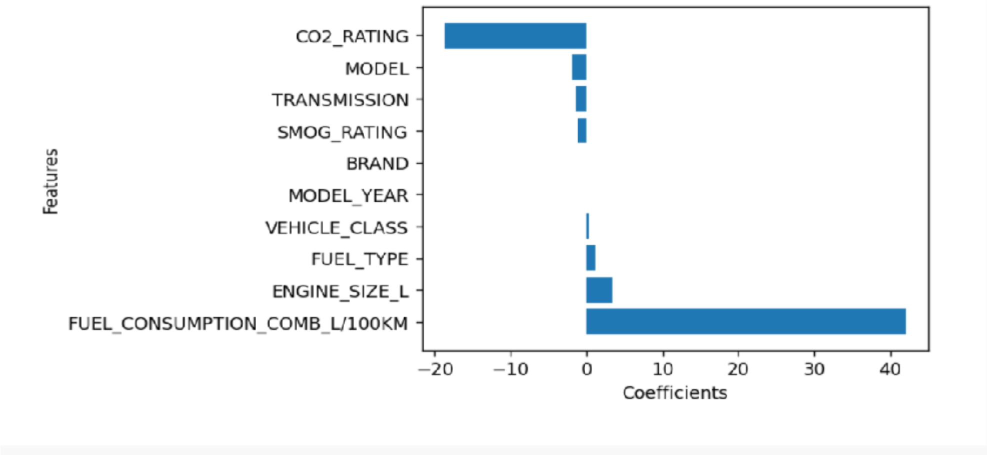


Fig 3.8

Case 3:

Independent variables:

MODEL_YEAR

MODEL

VEHICLE_CLASS

ENGINE_SIZE_L

CYLINDERS

TRANSMISSION

FUEL_TYPE

FUEL_CONSUMPTION_CITY_L/100KM

FUEL_CONSUMPTION_HWY_L/100KM

FUEL_CONSUMPTION_COMB_L/100KM

FUEL_CONSUMPTION_COMB_MPG

CO2_EMISSIONS

CO2_RATING

SMOG_RATING

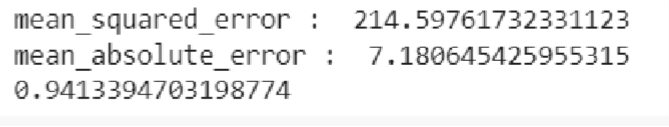
Dependent variable:

CO2_EMISSIONS

In case 3 we have only considered Co2 emissions as we wanted to test out the performance of a model in which the predictable feature was the only one to be considered for the evaluation of the model as this would give us a holistic understanding of the model's key performance metrics when the only variable to be considered is the one which is to predicted.

We have considered Stochastic Gradient Descent optimisation algorithm for this case as well

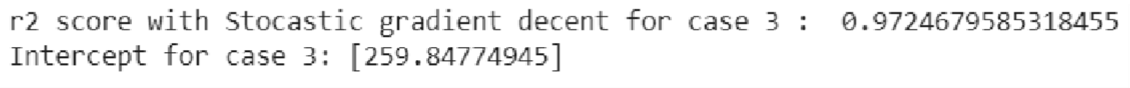
The result of the following study of linear regression with and without Stochastic Gradient Descent algorithm for Case 2 is:



```
mean_squared_error :  214.59761732331123
mean_absolute_error :  7.180645425955315
0.9413394703198774
```

Fig 3.9

Linear Regression without Stochastic Gradient Descent algorithm

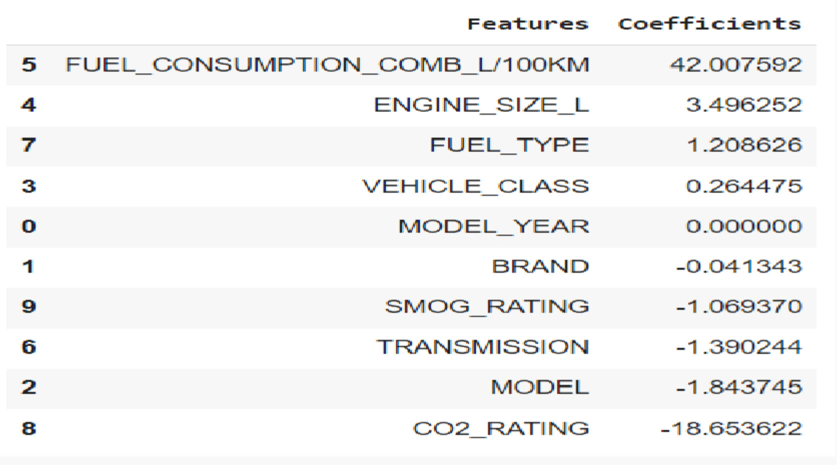


```
r2 score with Stocastic gradient decent for case 3 :  0.9724679585318455
Intercept for case 3: [259.84774945]
```

Fig 3.10

Linear Regression with Stochastic Gradient Descent algorithm

Features of the Stochastic Gradient Descent algorithm along with their coefficients are represented in this image



	Features	Coefficients
5	FUEL_CONSUMPTION_COMB_L/100KM	42.007592
4	ENGINE_SIZE_L	3.496252
7	FUEL_TYPE	1.208626
3	VEHICLE_CLASS	0.264475
0	MODEL_YEAR	0.000000
1	BRAND	-0.041343
9	SMOG_RATING	-1.069370
6	TRANSMISSION	-1.390244
2	MODEL	-1.843745
8	CO2_RATING	-18.653622

Fig 3.11

In the form of a graphical representation

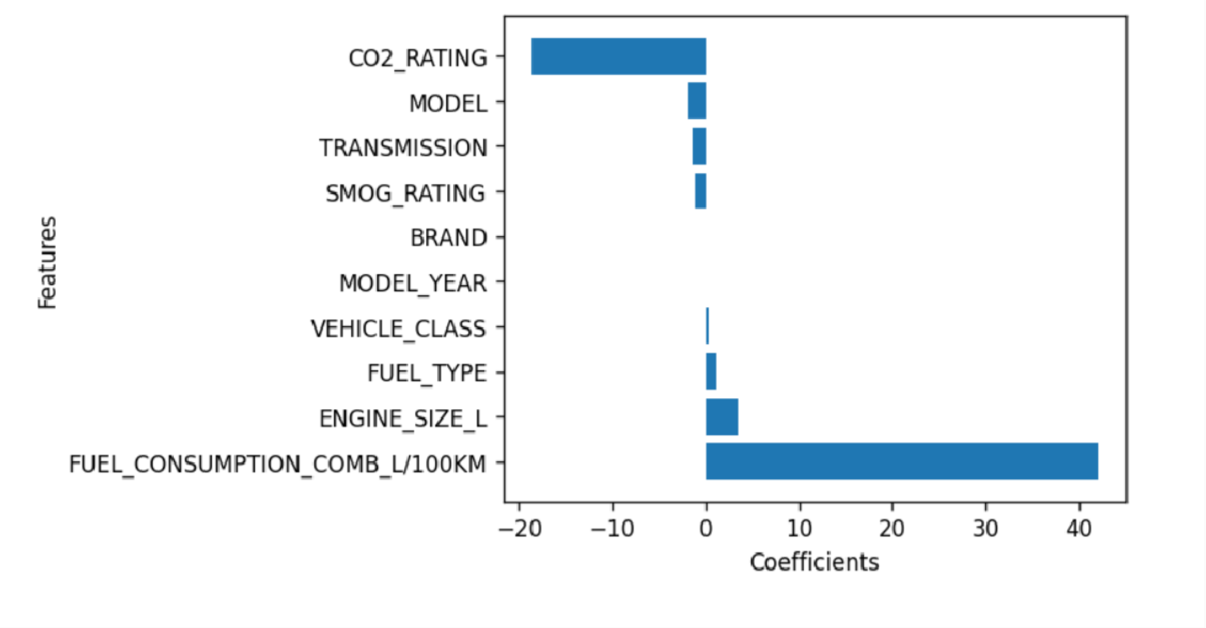
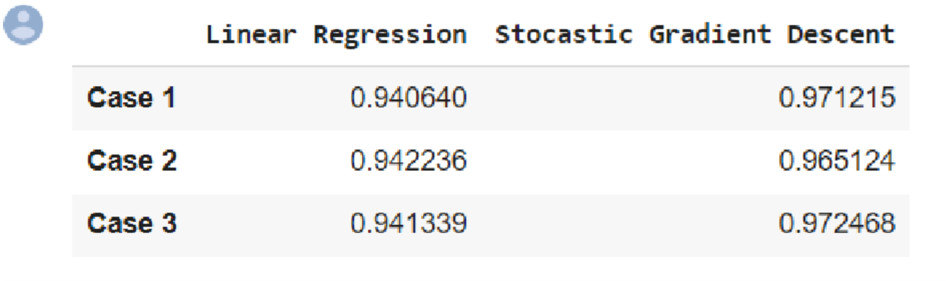


Fig 3.12

MODEL COMPARISION

Since we have three cases and, in all cases, we have applied the optimisation algorithm

The table down below gives us the performance of all the three cases with the different models



	Linear Regression	Stocastic Gradient Descent
Case 1	0.940640	0.971215
Case 2	0.942236	0.965124
Case 3	0.941339	0.972468

From the image it is clear that when it comes to Linear regression without the optimisation algorithm, Case 2 is the best performer

When it comes to Linear regression with Stochastic Gradient, Case 3 is marginally better than Case 1

Cumulatively we can conclude that Stochastic Gradient model of Case 1 is more applicable in the real world especially for when considering the magnitude of the dataset because the data at hand is dealing with the prediction of Carbon dioxide emission from the given set of data features, so with the number of features in Case 1 this particular model has performed immaculately, as the prediction of Co2 can be very vital for the vehicular industry and for the environment it is best to pick a model which has yielded a great result when considering there is only a limit to the features now in the dataset and many other features can be added to dataset which will help the industries to produce more environmental friendly vehicles in the future with less CO2 emissions because this particular algorithm has proven to yield favourable results with more no of features, hence predicting the future of the vehicular performance in Canada

FUTURE DEVELOPMENTS

There are usually some differences between the actual condition and the projection, which enables for the assessment of future trends. In certain research, the carbon dioxide emissions inside these borders are predicted using ML models for a city, a nation, or an enterprise. Most of these models combine several methods into a single model, such as Decision Trees or Neural Network Support Vector Machines (Saleh et al., 2016; Yang et al., 2018). (Kadam and Vijayumar, 2018). It is important to note that these models either use energy-related predictors or historical emission levels to forecast current values (energy consumption breakdown by fuel types or electricity consumption) [7]

REFERENCES

[1] https://www.datarobot.com/wiki/data-preparation/#:~:text=What%20is%20Data%20Preparation%20for,uncover%20insights%20or%20make%20predictions.

[2] https://www.sharpsightlabs.com/blog/pandas-isnull/

[3] https://www.kaggle.com/datasets/rinichristy/2022-fuel-consumption-ratings

[4] https://www.techtarget.com/searchbusinessanalytics/feature/Data-preparation-in-machine-learning-6-key-steps

[5] https://www.geeksforgeeks.org/python-pandas-isnull-and-notnull/

[6] https://statisticsbyjim.com/basics/normal-distribution/

[7] Kadam, Pooja & Vijayumar, Suhasini. (2018). Prediction Model: CO2 Emission Using Machine Learning. 1-3. 10.1109/I2CT.2018.8529498.

The following websites/platforms helped us with respect to the construction of the model (Code of the model)

https://scikit-learn.org/stable/

https://seaborn.pydata.org/

https://matplotlib.org/

https://www.kaggle.com/

https://towardsdatascience.com/

https://stackoverflow.com/

https://www.youtube.com/