CAR PERFORMANCE PREDICTION

Using Machine Learning

Developed by:Kavya Sriram,Madhu Dugyala,Shravani Ambaragonda,Veeranna kathri

Smart Bridge - Remote Summer Internship Program

1.INTRODUCTION

Car performance (mpg) prediction is somehow interesting and popular problem. Accurate car performance prediction involves expert knowledge, because performance usually depends on many distinctive features and factors. Typically, most significant once are mpg (miles per gallon), horsepower, engine and peak rpm. Here, we applied different methods and techniques in order to achieve the higher performance of the car.

Machine learning is a branch in computer science that studies the design of algorithms that can learn. Typical machine learning tasks are concept learning, function learning or predictive modeling, clustering and finding predictive patterns. These tasks are learned through the available data that were observed through the experiences or instructions. For example, Machine learning hopes that including the experience into its tasks will eventually improve the learning. The ultimate goal is to improve the learning in such a way that it becomes automatic, so that humans like ourselves don't need to interfere any more. Machine learning is a type of artificial intelligence in which the focus is on the development of computer programs that can learn from data and adapt to changes.

The most important goal for prediction involves using some variables in dataset to predict unknown values of other variable. There are huge number of phases in the prediction based on Machine Learning. Data collection is the first phase, by this phase data should be collected not usually a less data set, it should be huge data set according to the requirements one should collect or create the data for the prediction. Data Preprocessing is the second phase and this contain a lot of sub-phases for the processing of the data, and it includes importing libraries, Data Visualization, Data Transformation, Feature Scaling, Splitting and Label Encoding. Data Splitting, in this phase the data is to be split into two as train- data and test-data for the training of the model. Then the Fourth phase is Model Training, Supervised learning allows for the processing data with target attributes or labeled data. These attributes are mapped in

historical data before the training begins. And the last phase is Model evaluation and Testing and it is to be develop the simplest model able to formulate a target value fast and well enough. A data scientist can achieve this goal through model tuning. That's the optimization of model parameters to achieve an algorithm's best performance. Machine Learning techniques helps in providing accurate mpg to each car. Our model in this data set will be trained on many different cars, and it should give us a good estimate for our unknown car's mpg. Note that some of the values in the dataset are incorrect, so we will be fixing those values as we preprocess the data. The most most important features in determining the performance of cars are found to be the engine horsepower, the engine type, engine cylinder are essential since they determine how well a car runs .

1.1Overview:

The particular data set we worked with is made up of 205 unique vehicles. Each entry contains unique information such as cylinders, horsepower, peak rpm, etc. This modelhelps in predicting mpg of a car based on different variables using Linear Regression. This analysis should also help's customers who wants to buy cars with good performance. Data Mining is one of the most motivating and vital area of research with aim of extracting information from tremendous amount of accumulated data sets. Here, a new model for classifying mpg performances in car's by using the Machine Learning concepts . This model has been built using data from Regional Transport Office to predict status of mpg. Three algorithms have been used to build the proposed model: Linear Regression, Random Forest, Decision Tree. By using this algorithm's a Flask model has been implemented and tested. The results as been discussed and a full comparison between algorithms was conducted.LinearRegression was selected as best algorithm based on accuracy.

1.2 Purpose:

In this project, we make use of pandas, numpy, matplotlib, and seaborn libraries and using packages such as scikit-learn to predict the mpg of the car. The automotive industry is extremely competitive. With increasing fuel prices and picky consumers, the automobile makers are constantly optimizing their processes to increase the fuel efficiency. And at the end,predicting the performance of the car using the techniques of Machine Learning algorithms and then withdrawing the conclusions.

2.LITERATURESURVEY

When it comes to estimating car mpg using the Machine Learning, we estimated car mpg using classical Machine Learning algorithms namely, Linear Regression, Decision Tree and Random Forest, with the highest accuracy 95% with Linear regression. Machine Learning is the process of analyzing data from different perspectives and extracting useful knowledge from it. It is the core of knowledge discover process.

The car performance prediction is particularly well suited to Regression Algorithm. In Regression, the independent variable's are so well trained to predict the dependent variable. And the model is evaluated.

2.1Existing Problem:

It's obvious that the mileage of a car doesn't depend purely on only some features. There are several other factors in play like direction and strength of wind, city roads , city traffic , and weather, driver experience and ability etc. .There are some issues that can low down the car performance , but there are a few features that are considered to be the leading causes.

2.2 Proposed Solution:

Machine Learning (Linear Regression)

The previous models have less accuracy and the predictions are not at accurate whereas this model is constrained with the lot of advantages and with higher accuracy than any other model already proposed. In this model we used Machine Learning algorithm named Linear Regression which give an accuracy 95% of the predicted problem and there is an user friendly user interface to check the mpg for the cars.

Linear Regression is one of the simplest and most common Supervised Machine Learning algorithms that data scientists use for predictive modelling. We'll use the Linear Regression to build a model that predicts the performance(mpg) of the car.And also we have created an UI using the Flask for the performance prediction and this UI will allow the users to predict their performance of the car very easily and the User interface is user friendly not at least one complication in using the interface, and it can be used just by entering some necessary details into the UI in real time it'll give the predicted value like the mpg of the cars whether they give the best

performance or not.

3. THEORITICAL ANALYSIS

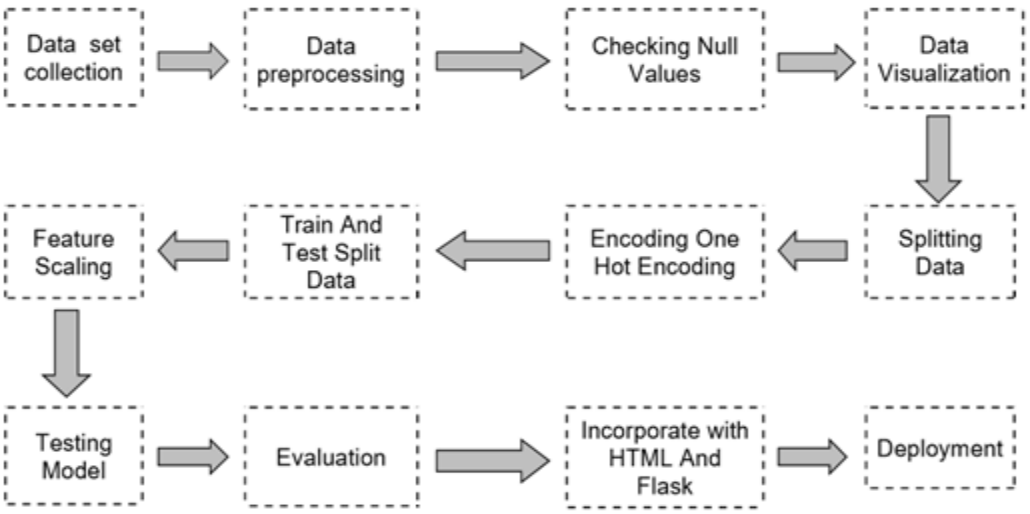
While selecting the algorithm that gives an accurate prediction we gone through lot of algorithms which gives the results abruptly accurate and from them we selected only one algorithm for the prediction problem that is Linear Regression. It gives the output based on the independent variables accurately, that's how the prediction work great with the Linear Regression Algorithm.

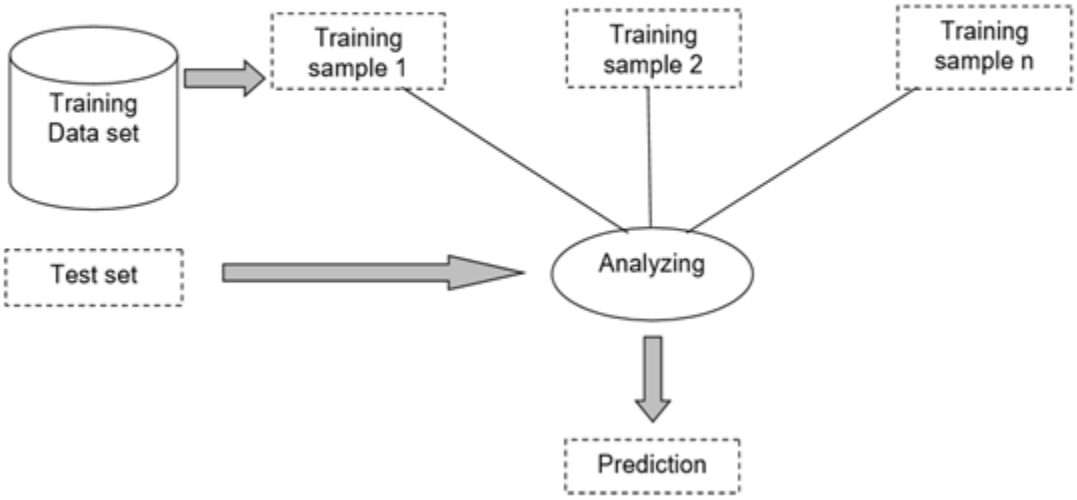
The peculiarity of this problem is collecting the data from mpg of car details in real time and working with the prediction at the same time, so we developed an user interface for the customer's who'll be accessing for the car's performance prediction. Accuracy is defined as the ratio of the number of sample correctly classified by the classifier to the total number of samples for a given test data set. The formula is as follows :

Accuracy=TP+TN/TP+TN+FT+FN

At first, we got lot of worst accuracies because we tried lot of given algorithms for the best accurate algorithm , finally after all of that we tried the best suitable algorithm which gives the prediction accurately is the Linear Regression algorithm. And developed it to use as a real time prediction problem for the car performance(mpg) prediction.

3.1 Block Diagram


Data set collection
Data preprocessing
Checking Null Values
Data Visualization
Splitting Data
Encoding One Hot Encoding
Train And Test Split Data
Feature Scaling
Testing Model
Evaluation
Incorporate with HTML And Flask
Deployment


Training Data set
Training sample 1
Training sample 2
Training sample n
Test set
Analyzing
Prediction

3.2 Software Designing:

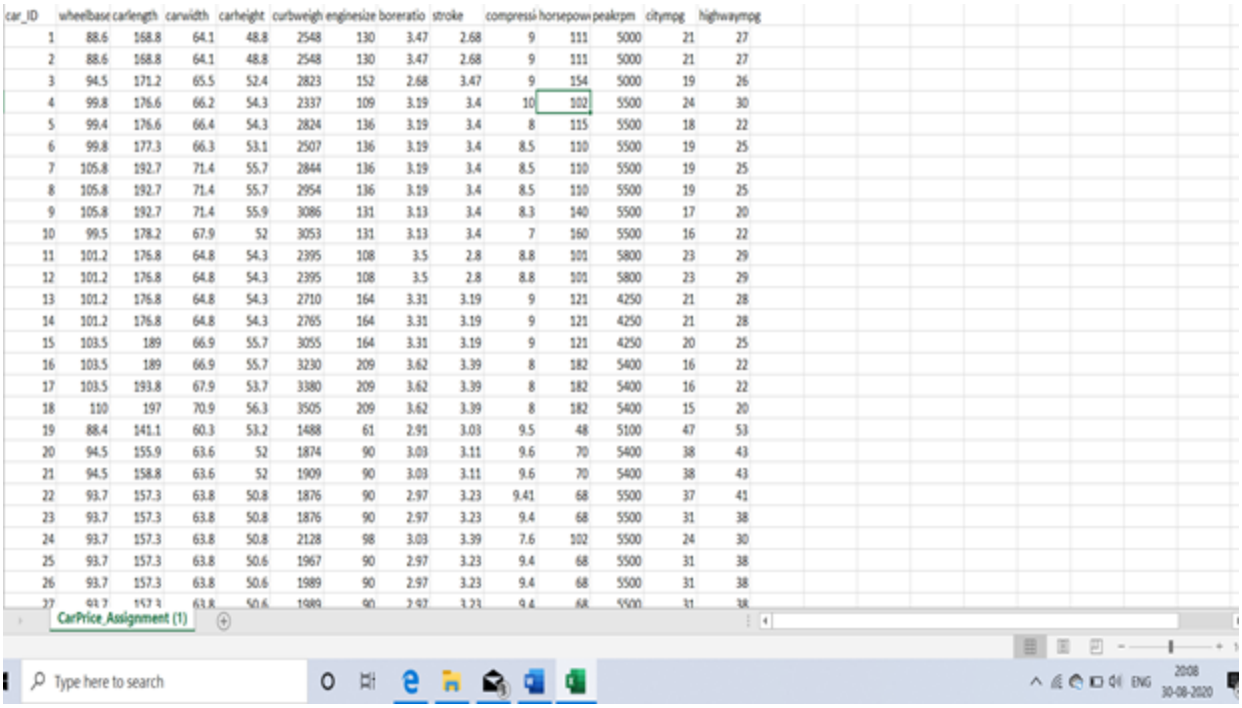
- Jupyter Notebook Environment
- Spyder Ide
- Machine Learning Algorithms
- Python (pandas, numpy, matplotlib, seaborn, sklearn)
- HTML
- Flask

We developed this car performance prediction by using the Python language which is a interpreted an high-level programming language and using the Machine Learning algorithms. For coding, weused the Jupyter Notebook environment of Anaconda distributions and the Spyder, it is an integrated scientific programming in the python language.

For creating an user interface for the prediction we used the Flask. It is a micro web framework written in the Python. It is classified as a micro framework because it does not require any particular tools or the libraries. It has no data base abstraction layer, form validation, or any other components where pre-existing third-party libraries which provide common functions, and a scripting language to create a webpage is HTML by creating the templates to use in the functions of the Flask and HTML.

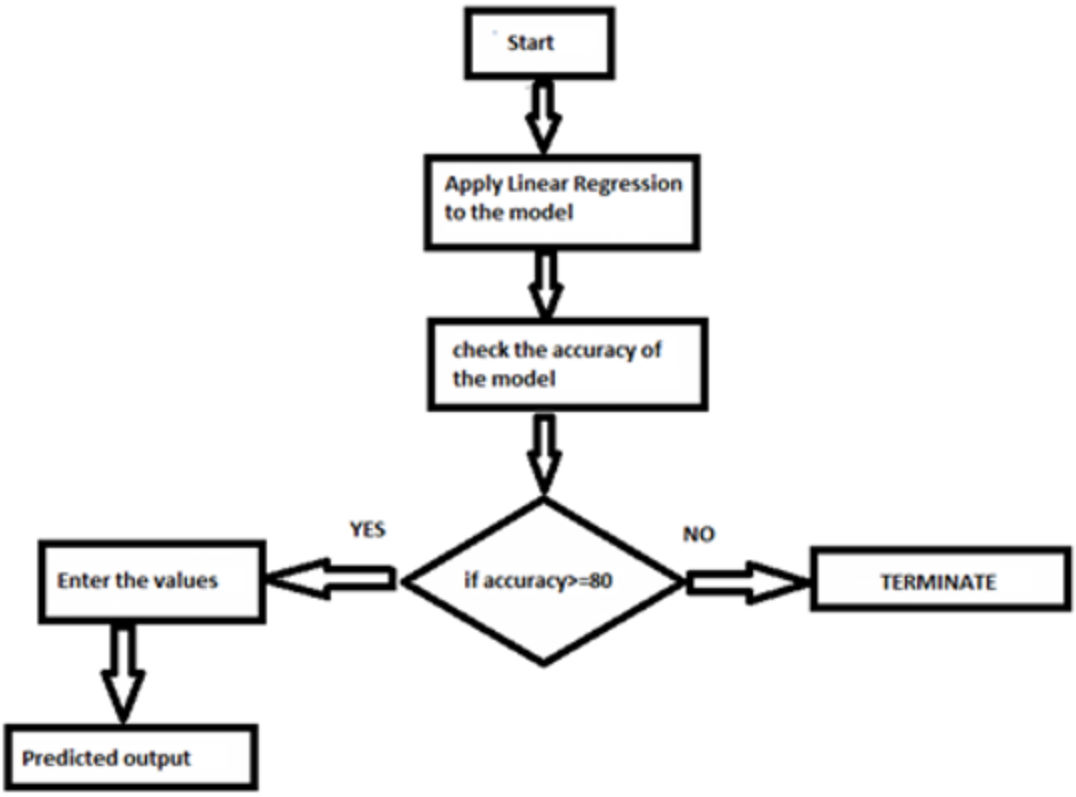
4. EXPERIMENTAL INVESTIGATION

In this paper, the data set we used is derived from www.kaggle.com . It contains 1992 original data of car mpg with 26 attributes. After that, the missing values are filled in by means of mode interpolation, and the duplicate or meaningless attributes are deleted, finally we have retained to 14 attributes. Those attributes were shown below in the screenshot of the data set we used.



car_ID	wheelbase	carlength	carwidth	carheight	curbweigh	enginesize	boreratio	stroke	compressi	horsepow	peakrpm	citympg	highwaympg
1	88.6	168.8	64.1	48.8	2548	130	3.47	2.68	9	111	5000	21	27
2	88.6	168.8	64.1	48.8	2548	130	3.47	2.68	9	111	5000	21	27
3	94.5	171.2	65.5	52.4	2823	152	2.68	3.47	9	154	5000	19	26
4	99.8	176.6	66.2	54.3	2337	109	3.19	3.4	10	102	5500	24	30
5	99.4	176.6	66.4	54.3	2824	136	3.19	3.4	8	115	5500	18	22
6	99.8	177.3	66.3	53.1	2507	136	3.19	3.4	8.5	110	5500	19	25
7	105.8	192.7	71.4	55.7	2844	136	3.19	3.4	8.5	110	5500	19	25
8	105.8	192.7	71.4	55.7	2954	136	3.19	3.4	8.5	110	5500	19	25
9	105.8	192.7	71.4	55.9	3086	131	3.13	3.4	8.3	140	5500	17	20
10	99.5	178.2	67.9	52	3053	131	3.13	3.4	7	160	5500	16	22
11	101.2	176.8	64.8	54.3	2395	108	3.5	2.8	8.8	101	5800	23	29
12	101.2	176.8	64.8	54.3	2395	108	3.5	2.8	8.8	101	5800	23	29
13	101.2	176.8	64.8	54.3	2710	164	3.31	3.19	9	121	4250	21	28
14	101.2	176.8	64.8	54.3	2765	164	3.31	3.19	9	121	4250	21	28
15	103.5	189	66.9	55.7	3055	164	3.31	3.19	9	121	4250	20	25
16	103.5	189	66.9	55.7	3230	209	3.62	3.39	8	182	5400	16	22
17	103.5	193.8	67.9	53.7	3380	209	3.62	3.39	8	182	5400	16	22
18	110	197	70.9	56.3	3505	209	3.62	3.39	8	182	5400	15	20
19	88.4	141.1	60.3	53.2	1488	61	2.91	3.03	9.5	48	5100	47	53
20	94.5	155.9	63.6	52	1874	90	3.03	3.11	9.6	70	5400	38	43
21	94.5	158.8	63.6	52	1909	90	3.03	3.11	9.6	70	5400	38	43
22	93.7	157.3	63.8	50.8	1876	90	2.97	3.23	9.41	68	5500	37	41
23	93.7	157.3	63.8	50.8	1876	90	2.97	3.23	9.4	68	5500	31	38
24	93.7	157.3	63.8	50.8	2128	98	3.03	3.39	7.6	102	5500	24	30
25	93.7	157.3	63.8	50.6	1967	90	2.97	3.23	9.4	68	5500	31	38
26	93.7	157.3	63.8	50.6	1989	90	2.97	3.23	9.4	68	5500	31	38

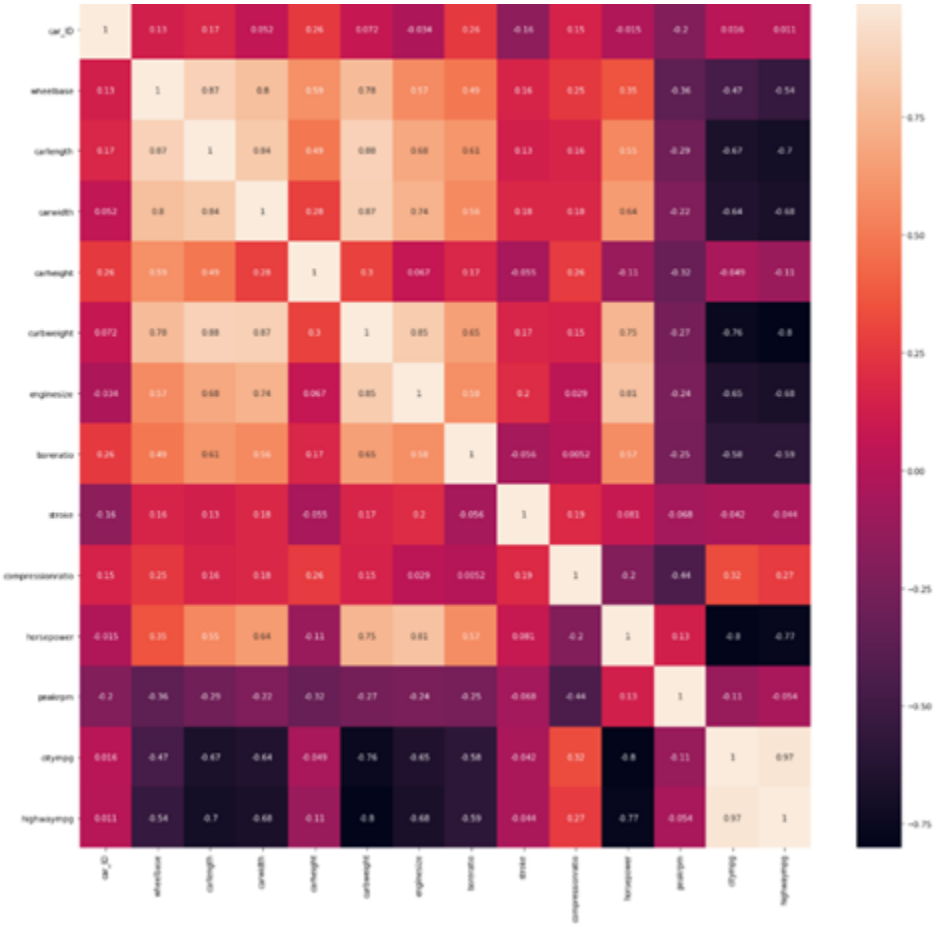
5. FLOWCHART

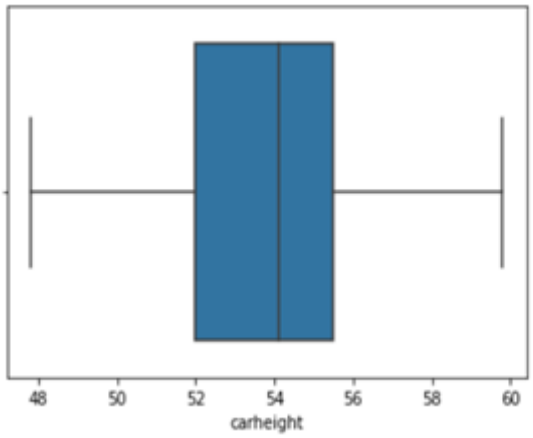


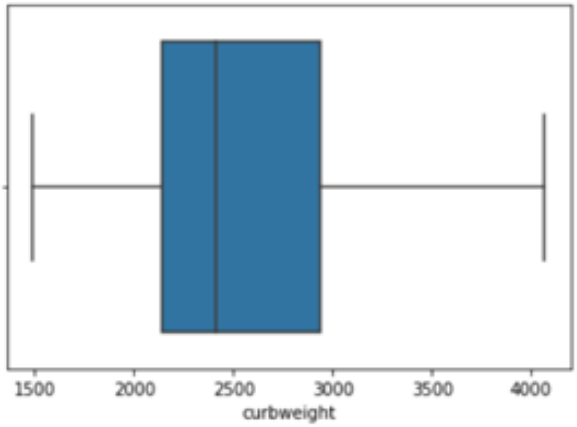
6.RESULT

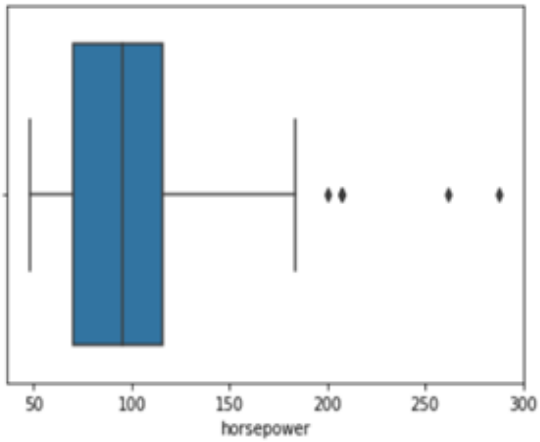
In this paper, the Linear Regression algorithm is used to predict its performance,and compared with another two Machine Learning methods namely the Decision Tree, the Random Forest. The obtained results are displayed in the Table below. The results show that, the performance of the Linear Regression have better performance than Random Forest and Decision Tree. The Linear Regression is best with an accuracy of 95% higher than Random Forest with an accuracy of 86%.

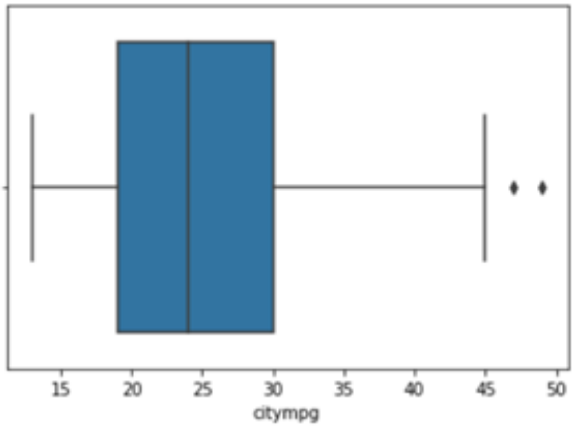
The given are the heatmap of the dataset which represents the correlation between the attributes and the boxplot of each attribute.

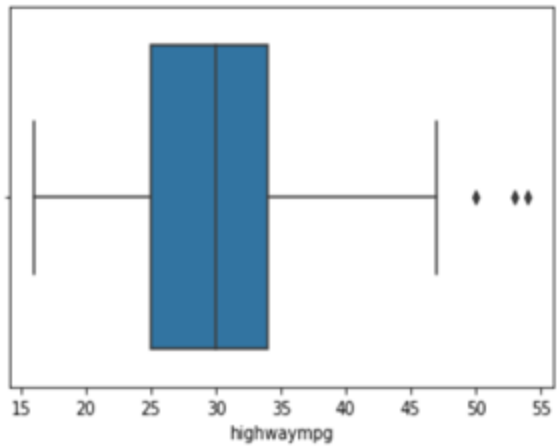

car_ID
wheelbase
carlength
carwidth
carheight
curbweight
enginesize
boreratio
stroke
compressionratio
horsepower
peakrpm
citympg
highwaympg
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75


48
50
52
54
56
58
60
carheight

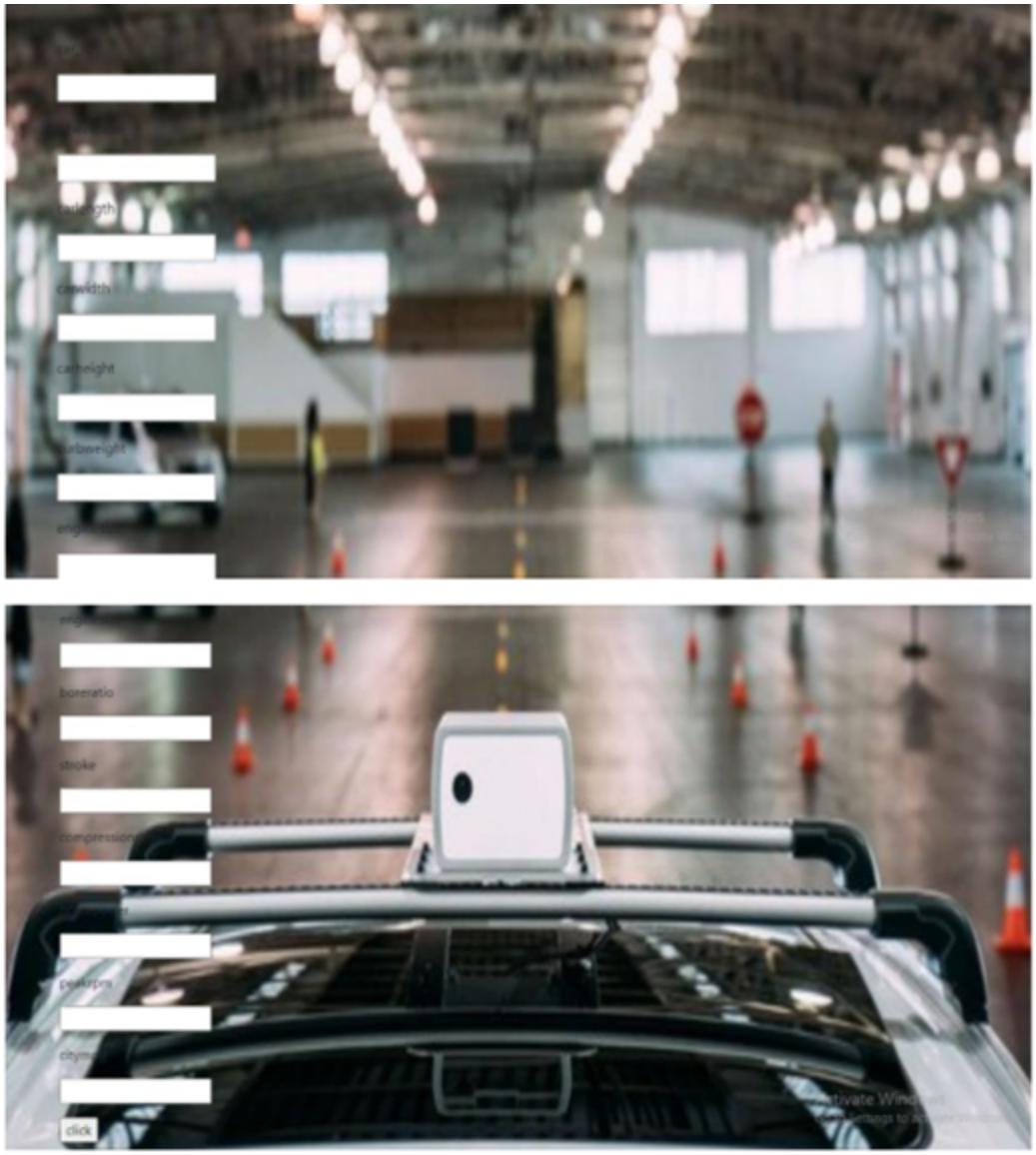

1500
2000
2500
3000
3500
4000
curbweight


50
100
150
200
250
300
horsepower


15
20
25
30
35
40
45
50
citympg



Serial no.	Algorithms Used	Accuracy
1.	Linear Regression	0.95%
2.	Decision Tress	0.86%
3.	Random Forest	0.93%



7. ADVANTAGES AND DISADVANTAGES

Advantages:

• Effective predictive model which predicts whether car mpg is “High” or “Low”.

• Reduce Oil Dependency.

• Easy and simple User Interface for the customer’s who is going to evaluate the performance of the car.

• Linear Regression is simple to implement and easy to interprete the output of the

coefficient.

• Linear Regression gives the accurate result of the prediction upto 95% which is the algorithm we used for prediction.

• It is composed using the HTML and Python for the web usage in real time.

• It can work in real time and predicted as soon as the necessary details for prediction are given to the model.

Disadvantages:

•The model should be highly predictive in nature i.e it should show 80% of accuracy.

•The model should give high accuracy when tested it on the test dataset.

• One of the biggest drawback is using gasoline –powered cars in the amount of pollution into the atmosphere .

• On the other hand in the linear regression technique outliers can have huge effects on the regression and boundaries are linear in this technique.

8.APPLICATIONS

• Predicting of a car mileage definitely help an individual seeking for a car that can be trusted and valued.

• This model is mostly used for customers who wants to buy car's with best mileage performance.

• To have an idea of customer relationship cycle such as customer acquisition, increasing value of the customer and customer retention.

• So we use Machine Learning Algorithms to predict the performance of the car.

9. CONCLUSION

Car performance prediction can be a challenging task due to the high number of attributes that should be considered for the accurate prediction. During this model, we built a model that could reliably predict a car's mpg given some information about the car within 27 mpg of the actual value. This model could be trained with car data and be used to predict competitor's future mpg ratings for upcoming cars.

In this paper, the Linear Regression algorithm is adopted to build a UI model for predicting performance of the car and the results are compared with other two

algorithms of Random Forest and Decision Tree. This model shows that Linear Regression performs best than the other two algorithms in the prediction of mpg. There is no definitive guide of which algorithms to be used. What may work on some data sets may not work on others. Therefore, always check the accuracy and predict with the dataset values.

10. FUTURESCOPE

This Linear Regression model can also be used in the future predictions like weather forecast, job prediction, salary prediction etc. In future we can create a web application on this type of problems so that it can be accessed from everywhere with more users.

In future, the Linear Regression algorithm can be applied on other data sets available for car mpg to further investigate its accuracy. A rigorous analysis of other Machine Learning algorithms other than this can also be done in future to investigate the power of machine learning algorithms for mpg prediction.

In further study, we will try to conduct experiments on larger data sets or try to tune the model so as to achieve the state of art performance of the model and a great UI support system making it complete web application model.

11. BIBILIOGRAPHY

●"The data concerns cars performance in miles per gallon, to be predicted in terms of 14 attributes." (Quinlan, 1993)

●Quinlan,R.(1993).Combining Instance-Based and Model-Based Learning.In the Proceedings on the Tenth International Conference of Machine Learning, 236-243, University of Massachusetts, Amherst. Morgan Kaufmann.

●Dan Pelleg. Scalable and Practical Probability Density Estimators for Scientific Anomaly Detection. School of Computer Science Carnegie Mellon University. 2004. [View Context].

●Christopher R. Palmer and Christos Faloutsos. Electricity Based External Similarity ofAttributes.PAKDD.2003.[ViewContext].

●Thomas Melluish and Craig Saunders and Ilia Nouretdinov and Volodya Vovk and Carol S. Saunders and I. Nouretdinov V..The typicalness framework: a comparison with the Bayesian approach.Department of Computer Science. 2001. [View Context].

●Wai Lam and Kin Keung and Charles X. Ling. PR 1527. Department of Systems Engineering and Engineering Management, The Chinese University of Hong Kong. 2001.[ViewContext].
●Dan Pelleg and Andrew W. Moore.Mixtures of Rectangles: Interpretable Soft Clustering.ICML.2001.[ViewContext].
●Jinyan Li and Kotagiri Ramamohanarao and Guozhu Dong.. PAKDD. 2001. [View Context]. ●Zhi-Hua Zhou and Shifu Chen and Zhaoqian Chen.. IJCNN (3). 2000. Mauro Birattari and Gianluca Bontempi and Hugues Bersini.NIPS. 1998. [View Context].

APPENDIX

HTML:

```
<htmllang="en">

<head>

  <metacharset="UTF-8">

  <metaname="viewport" content="width=device-width, initial-scale=1.0">

  <metahttp-equiv="X-UA-Compatible"content="ie=edge">

  <title>Car performance prediction</title>

                <linkhref="https://cdn.bootcss.com/bootstrap/4.0.0/css/bootstrap.min.css"
rel="stylesheet">

  <script src="https://cdn.bootcss.com/popper.js/1.12.9/umd/popper.min.js"></script>

  <script src="https://cdn.bootcss.com/jquery/3.3.1/jquery.min.js"></script>

  <script src="https://cdn.bootcss.com/bootstrap/4.0.0/js/bootstrap.min.js"></script>

  <link href="{{ url_for('static', filename='css/main.css') }}" rel="stylesheet">

</head>

<body
style="background-image:url('https://www.autocar.co.uk/sites/autocar.co.uk/files/styles/body-image/public/1-kia-stinger-gt-s-hero-front.jpg?itok=cZ527kzA'); background-size: 100% 100%;">

 <navclass="navbar navbar-dark bg-dark">

    <div class="container">
```

```
        <aclass="navbar-brand" href="#">Car Performance Prediction</a>
      </div>
    </nav>
    <div class="container">
      <div id="content" style="margin-top:2em">
<div class="container">
 <div class="row">
<div class="col-sm-6 bd">
<h3><font color="white">Car Performance Prediction</font></h3>
<br>
<formaction ="/login" method ="post">
<p>car_ID</p>
<p><input type ="number" name ="car_ID"/></p>
<p>wheelbase</p>
<p><inputtype ="float" name ="wheelbase"/></p>
<p>carlength</p>
<p><input type ="float" name ="carlength"/></p>
<p>carwidth</p>
<p><input type ="float" name ="carwidth"/></p>
<p>carheight</p>
<p><input type ="float" name ="carheight"/></p>
<p>curbweight</p>
<p><input type ="number" name ="curbweight"/></p>
<p>enginesize</p>
<p><input type ="number" name ="enginesize"/></p>
<p>boreratio</p>
<p><inputtype ="float" name ="boreratio"/></p>
<p>stroke</p>
<p><inputtype ="float" name ="stroke"/></p>
```

```
<p>compressionratio</p>
<p><inputtype ="float"name ="compressionratio"/></p>
<p>horsepower</p>
<p><inputtype ="number"name ="horsepower"/></p>
<p>peakrpm</p>
<p><inputtype ="number"name ="peakrpm"/></p>
<p>citympg</p>
<p><inputtype ="number"name ="citympg"/></p>
<p><inputtype ="submit" value ="click"/></p>
<b>{{y}}</b>
</div>
 </div>
</div>
 </div>
 </div>
  </div>
</body>
<footer>
 <scriptsrc="{{ url_for('static', filename='js/main.js') }}" type="text/javascript"></script>
</footer>
</html>
```

APP .PY:

```
from flask import Flask,render_template,request
import pickle
import pickle,joblib
model=pickle.load(open('RandomForestRegressor.pkl','rb'))
 app=Flask(__name__)
@app.route('/')
```

```
def hello_world():
    return render_template("base.html")
@app.route('/login',methods=["POST"])
def func2():
    car_ID=request.form['car_ID']
    wheelbase=request.form['wheelbase']
    carlength=request.form['carlength']
    carwidth=request.form['carwidth']
    carheight=request.form['carheight']
    curbweight=request.form['curbweight']
    enginesize =request.form['enginesize']
    boreratio=request.form['boreratio']
    stroke=request.form['stroke']
    compressionratio=request.form['compressionratio']
    horsepower=request.form['horsepower']
    peakrpm=request.form['peakrpm']
    citympg=request.form['citympg']

data=[[int(car_ID),float(wheelbase),float(carlength),float(carwidth),float(carheight),in
t(curbweight),int(enginesize),float(boreratio),float(stroke),float(compressionratio),int(
horsepower),int(peakrpm), int(citym pg)]]
pred=model.predict(data)
print(pred[0])
return render_template("base.html",y= 'CAR MPG is '+str(pred))
if __name__=='__main__':
  app.run(debug= True)
```

jbjghjghj