CAR RESALE VALUE PREDICTION

A PROJECT REPORT

Submitted by

SELVAPRIYA M	-	953119104039
SIVAKAMI M	-	953119104041
SUMATHI N	-	953119104043
VIJAYA A	-	953119104049

in partial fulfillment for the award of the degree

of

BAHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

THAMIRABHARANI ENGINEERING COLLEGE, TIRUNELVELI

ANNA UNIVERSITY::CHENNAI 600 025

NOVEMBER 2022

TABLE OF CONTENTS

1. INTRODUCTION

1.1 Project Overview:

With difficult economic conditions, it is likely that sales of second-hand imported (reconditioned) cars and used cars will increase. In many developed countries, it is common to lease a car rather than buying it outright. After the lease period is over, the buyer has the possibility to buy the car at its residual value, i.e. its expected resale value. Thus, it is of commercial interest to sellers/financers to be able to predict the salvage value (residual value) of cars with accuracy. In order to predict the resale value of the car, we proposed an intelligent, flexible, and effective systemthat is based on using regression algorithms. Considering the main factors which would affect the resale value of a vehicle a regression model is to be built that would give the nearest value of the vehicle. The data about car sales are derived from various sources. Then it will be integrated to the web-based application where the user is notified with the status of his product.

1.2 Purpose

car resale value prediction helps the user to predict the resale value of the car depending upon various features like kilometers driven, fuel type, etc. This resale value prediction system is made for general purpose **to just predict the amount that can be roughly acquired by the user.**

2. Literature Servey

2.1 Existing system

- Many data mining methods and machine learning algorithm are widely used to estimate the price of cars. The primary flaw with current system is that more characteristics are required to predict the pricing of the automobile.
- To obtain the result of prediction, further comparison techniques must be utilised more successfully. Obtaining dataset information is really challenging in the existing systems.The data sets won't include information on the automobiles that were long period of inactivity.

2.2 REFERENCES

1. Praful Rane, Deep pandya, Dhawal Kotak, "USED CAR PRICE PREDICTION", International Research Journal of Engineering and Technology (IRJET), Volume:08, issue:04, Apr 2021.

Description:

The Regression Algorithms had used because they provide continuous value as an output . Because of which it will be possible to predict the actual price of a car rather than the price range of a car.

2. Ketan Agrahari, Ayush Chaubey, mamoor Khan, Manas Srivastava, "Car Price Prediction Using Machine Learning", INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN TECHNOLOGY, Volume: 8, Issue: 1, June 2021.

Description:

In this system is to predict the car price as per the date set of previous consumer data like engine capacity, distance traveled year of manufacture etc., The result of these linear regression algorithm based on machine learning will be analysed and based on the efficiency and accuracy of the car price.

3. Aditya Nikhade, Rohan Borde, "USED CAR PRICE PREDICTION AND LIFE SPAN", International Advanced Research Journal in Science, Engineering and Technology, Vol: 8, Issue: 12, December 2021.

Description:

The prediction of the price of used cars is supervised with machine learning techniques. The prediction are based on the dataset collected from Kaggle website . Different techniques like multiple linear regression analysis, decision trees and k-nearest neighbors had used .

4. Ashutosh Datt Sharma, Vibhor Shorma, Sahil Mittal, Gautam Jain, Sudha Narang,"PREDICTIVE ANALYSIS OF USED CAR PRICES USING MACHINE LEARNING", International Research Journal of Modernization in Engineering Technology and Science, Volume:03, Issue:06, June-2021.

Description:

Employing various Machine Learning Algorithms, a statistical model was build and based upon the given data and features set to estimate the price of used cars. After applying various regression algorithm on the model, they concluded Decision Tree algorithm was the best performer.

5. Dhwani Nimbark, Akshat Patel, Sejal Thakkar, "Car Resale Value Prediction System", International Research Journal of Engineering and Technology(IRJET), Volume: 08, Issue: 05, May 2021.

Description:

Different algorithms had used like Support Vector Regression, Logistic Regression, Random Forest Regression and Gradient Boosting Regression for developing Car resale value prediction systems by considering different features of the car.

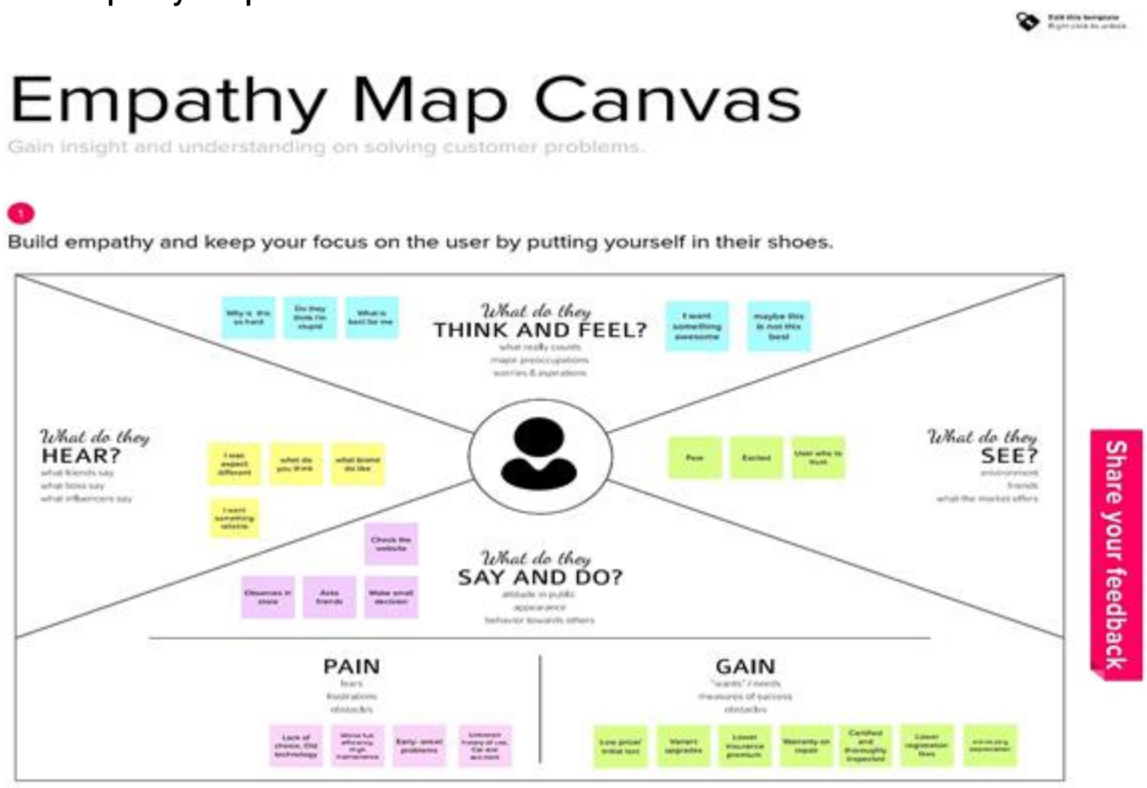
2.3 PROBLEM STATEMENT DEFINITION

Second hand car is predominantly used by all. So the prediction of used car price becomes the significant and interesting area of analysis. The price of the used car depends on the factors like miles driven, model and year of manufacturing.

Car resale value evaluator web application is developed to predict the price to attain benefits to buyer and seller.

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

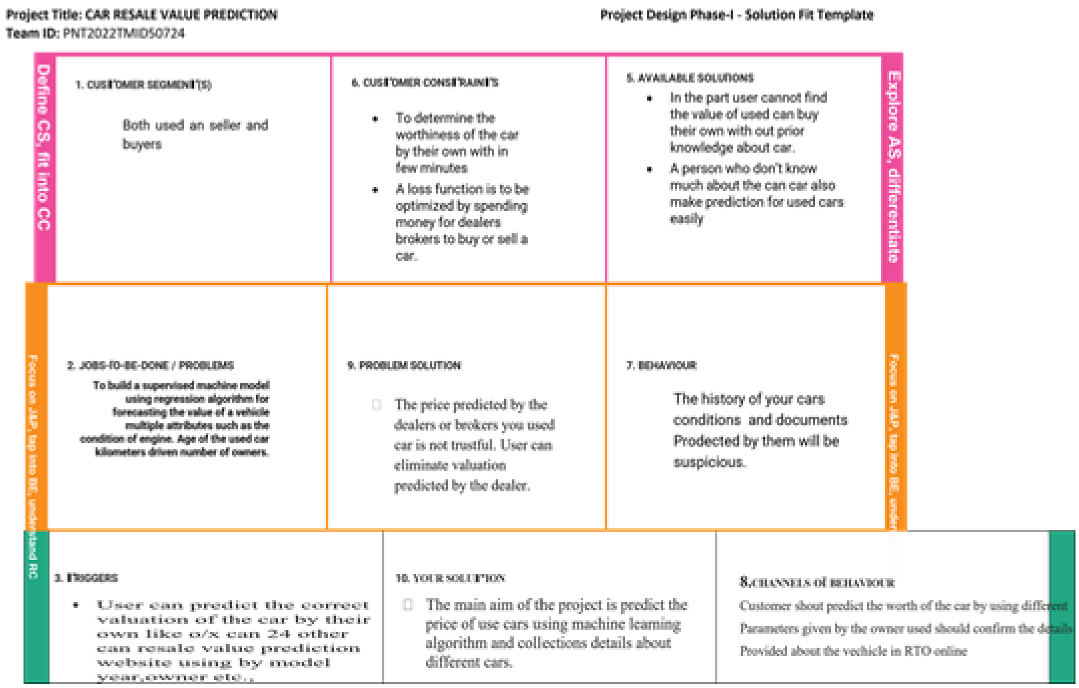


3.2 IDEATION & BRAINSTROMING

3.3 Proposed Solution

S. No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	This system also studies the service history and car usage statistics, without that we have no real idea of the condition of the car.
2.	Idea / Solution description	This process uses the accuracy score of regression algorithm for predicting the used car price.
3.	Novelty / Uniqueness	A centralized platform for car resale that will predict prices
4.	Social Impact / Customer Satisfaction	This analysis can be used to study the trends in the industry, offer better insight into the market.
5.	Business Model (Revenue Model)	In this model predominantly used by all the customer satisfaction of the model.
6.	Scalability of the Solution	The best accuracy of the linear regression algorithm will be taken as a solution.

3.4 Problem Solution Fit


Define CS, fit into CC
Explore AS, differentiate
Focus on J&P, tap into BE, understand RC
Focus on J&P, tap into BE, under

Project Title: CAR RESALE VALUE PREDICTION
Team ID: PNT2022TMID50724

Project Design Phase-I - Solution Fit Template

1. CUSTOMER SEGMENT(S)

Both used an seller and buyers

6. CUSTOMER CONSTRAINTS

- To determine the worthiness of the car by their own with in few minutes
- A loss function is to be optimized by spending money for dealers brokers to buy or sell a car.

5. AVAILABLE SOLUTIONS

- In the part user cannot find the value of used can buy their own with out prior knowledge about car.
- A person who don't know much about the can car also make prediction for used cars easily

2. JOBS-TO-BE-DONE / PROBLEMS

To build a supervised machine model using regression algorithm for forecasting the value of a vehicle multiple attributes such as the condition of engine. Age of the used car kilometers driven number of owners.

9. PROBLEM SOLUTION

□ The price predicted by the dealers or brokers you used car is not trustful. User can eliminate valuation predicted by the dealer.

7. BEHAVIOUR

The history of your cars conditions and documents Prodected by them will be suspicious.

3. TRIGGERS

- User can predict the correct valuation of the car by their own like o/x can 24 other can resale value prediction website using by model year,owner etc.,

10. YOUR SOLUTION

□ The main aim of the project is predict the price of use cars using machine learning algorithm and collections details about different cars.

8.CHANNELS of BEHAVIOUR

Customer shout predict the worth of the car by using different Parameters given by the owner used should confirm the details Provided about the vechicle in RTO online

4. Requirement Analysis

4.1 Functional Requirement

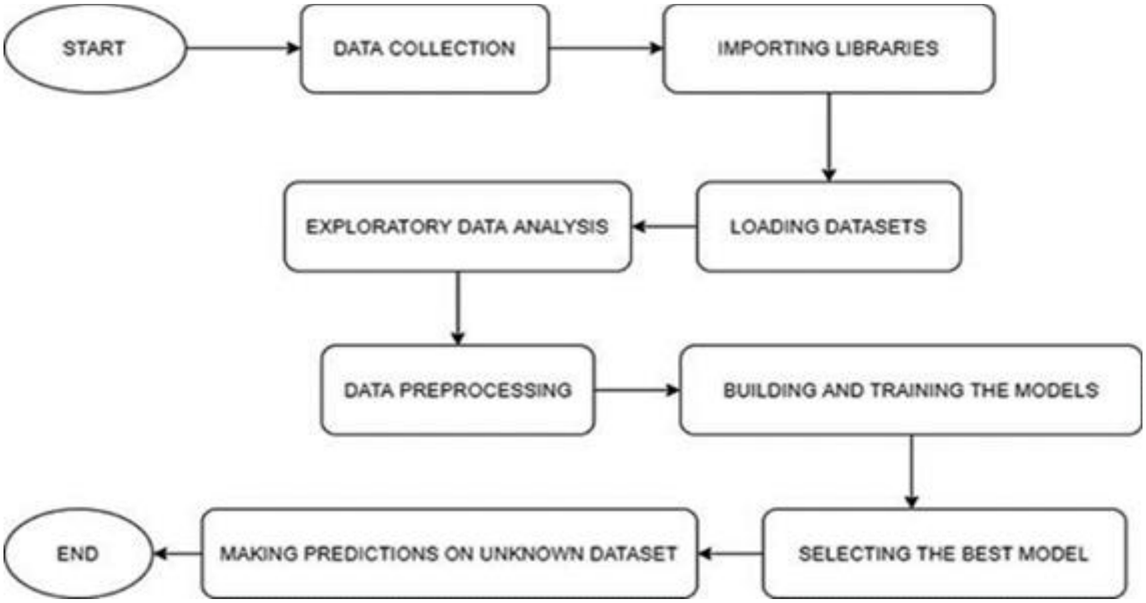
FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	Sign up and Login	Creation of an online account using username and password and used to gain acess to an area requiring proper authorization.
FR-2	Data Collection	Gathering quantitative and qualitative information on specific variables.
FR-3	Data Preprocessing	Preparing of raw data and making it suitable for a machine learning model.
FR-4	Test set	Trained on the initial training data set.
FR-5	Evaluation	Different evaluation metrics to understand a ML model's performance.
FR-6	Prediction	Output of an algorithm after it has been trained on a historical dataset.

4.2 Non- Functional Requirements

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	Ease of use and knows the details immediately.
NFR-2	**Security**	The application which protects the data efficiently over the web.
NFR-3	**Reliability**	The application can be used in a confidential manner.
NFR-4	**Performance**	The performance of the application is very affective.
NFR-5	**Availability**	Milage is available for the secondary car.
NFR-6	**Scalability**	This application is very scalable.

5. Project Design

5.1 Data Flow Diagrams



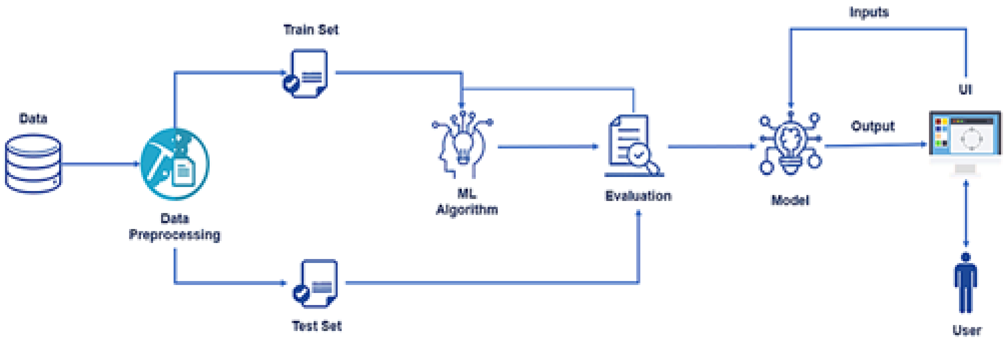
5.2 SOLUTION & TECHNICAL ARCHITECTURE

SOLUTION ARCHITECTURE

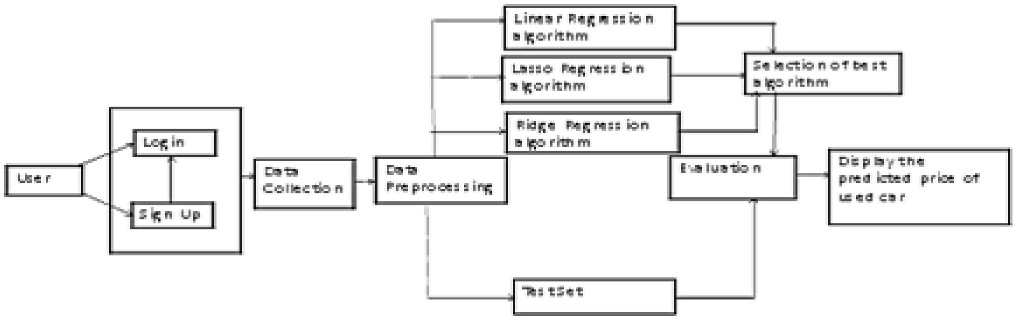
Solution architecture is a complex process – with many sub-processes – that bridges the gap between business problems and technology solutions. Its goals are to:

- Find the best tech solution to solve existing business problems.
- Describe the structure, characteristics, behavior, and other aspects of the software to project stakeholders.
- Define features, development phases, and solution requirements.
- Provide specifications according to which the solution is defined, managed, and delivered.

SOLUTION ARCHITECTURE DIAGRAM:



TECHNICAL ARCHITECTURE


User
Log in
Sign Up
Data Collection
Data Preprocessing
Linear Regression algorithm
Lasso Regression algorithm
Ridge Regression algorithm
Selection of best algorithm
Evaluation
Display the predicted price of used car
TestSet

5.3 USER STORIES

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority
User		USN-1	As a user, I can signup of an online accout using user name and password.	I can create user name and password	High
		USN-2	As a user, I can collecting the data of gathering the quantitative and qualitative information.	I can collect the data	High
		USN-3	As a user, I can register for the application through Gmail		Medium
customer	Login	USN-4	As a user, login the set of cerdentials and used to gain access to and area requring proper authorization		High

6. Project Planning & Scheduling

6.1 Sprint Planning & Estimation

Title	Description	Date
Literature survey & information Gathering	Literature survey on the selectedproject and gathering information by referring the technical papers, research publications	24 September 2022
Prepare Empathy Map	prepare empathy map canvas tocapture the user pains and gains, prepare list of problem statements	25 Semtember 2022
Ideation	List the by organizing the brainstroming session and prioritize the top 3 ideas based on the fesibility and importance	20 september 2022
Proposed solution	prepare the proposed solution doument, which includes thenovelty, feasibility of idea, business model, social impact, scalability of solution, etc	19 september 2022
Problem solution fit	prepare problem solution fit document	20 September 2022
Solution Architecture	prepare solution architecture document	19 September 2022

Customer Journey	Prepare the customer journey maps to understand the user interactions and experienes with the application	03 September 2022
Functional Requirement	prepare the funtional requirement document	03 October 2022
Data Flow Diagram	Draw the data flow dams and submit for review	03 October 2022
Technology Architeture	Prepare the technology architecture diagram	03 October 2022
Prepare Milestone and Activity List	prepare the milestones and activity list of the project	22 October 2022
Project Development Delivery of sprint-1, 2, 3, 4	Develop and submit the developed code by testing it	In progress.....

6.2 Sprint Delivery Schedule

Sprint	**Total Story Points**	**Duration**	**Sprint Start Date**	**Sprint End Data (Planned)**	**Story points completd (as on planned end date)**	**sprint release date (Actual)**
sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20	30 Oct 2022
sprint-2	20	6 Days	31 Oct 2022	5 Nov 2022	20	06 Nov 2022
sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	15 Nov 2022
sprint-4	20	6 Days	14 Nov 2022	14 Nov 2022	20	19 Nov 2022

7. PRIOR KNOWLEDGE

7.1 Data Collection

ML is a data hunger technology, it depends heavily on data, without data, it is impossible for a machine to learn. It is the most crucial aspect that makes algorithm training possible.

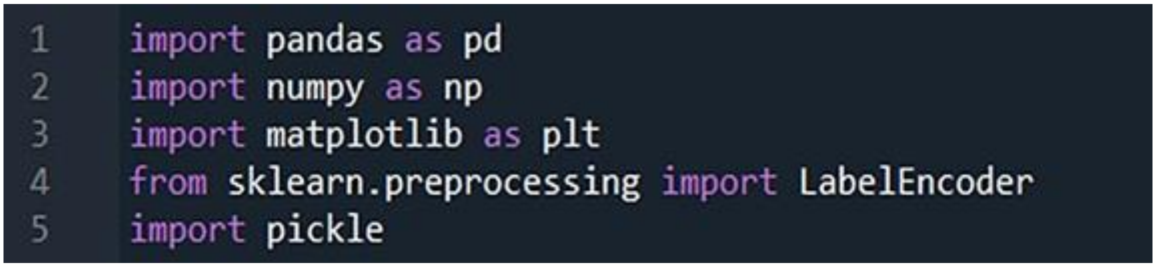
Pre-Process The Data

In this milestone, we will be preprocessing the dataset that is collected. Preprocessing includes:

i. Handling the null values.
ii. Handling the categorical values if any.
iii. Normalize the data if required.
iv. Identify the dependent and independent variables.
v. Split the dataset into train and test sets.

Import Reqired Libraries

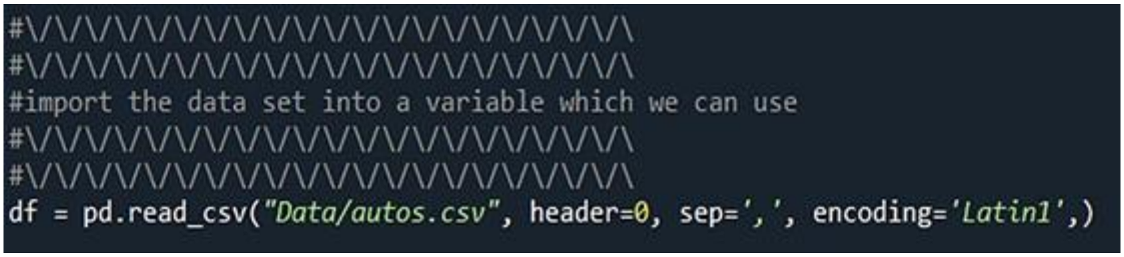
The libraries can be imported using the import keyword.



```python
import pandas as pd
import numpy as np
import matplotlib as plt
from sklearn.preprocessing import LabelEncoder
import pickle
```

Read The Datasets

The dataset is read as a data frame (df in our application) using the pandas library (pd is the alias name given to the pandas package).

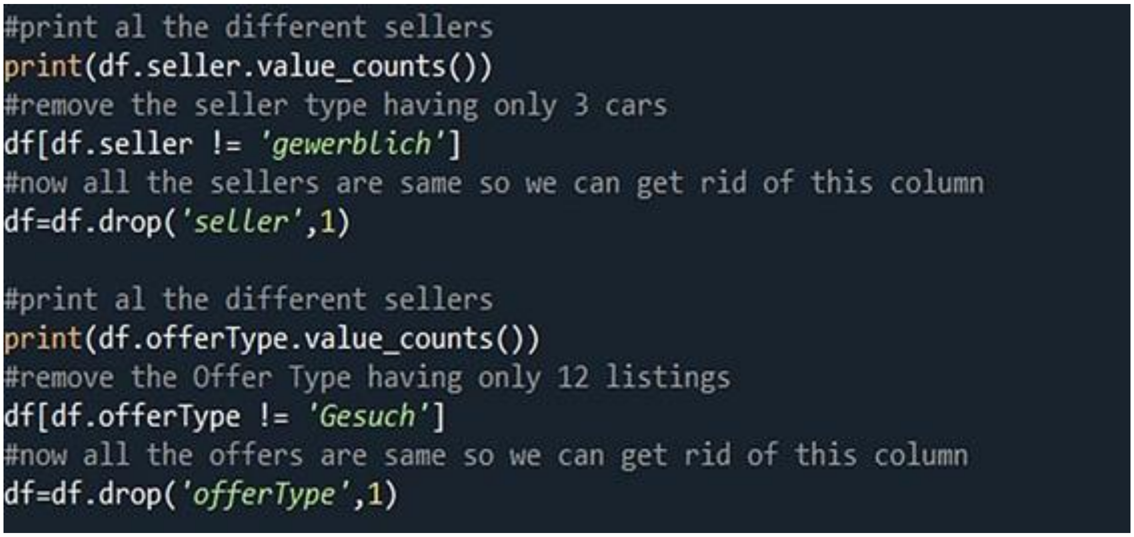


```python
#\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\
#\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\
#import the data set into a variable which we can use
#\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\
#\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\
df = pd.read_csv("Data/autos.csv", header=0, sep=',', encoding='Latin1',)
```

Cleaning The Dataset

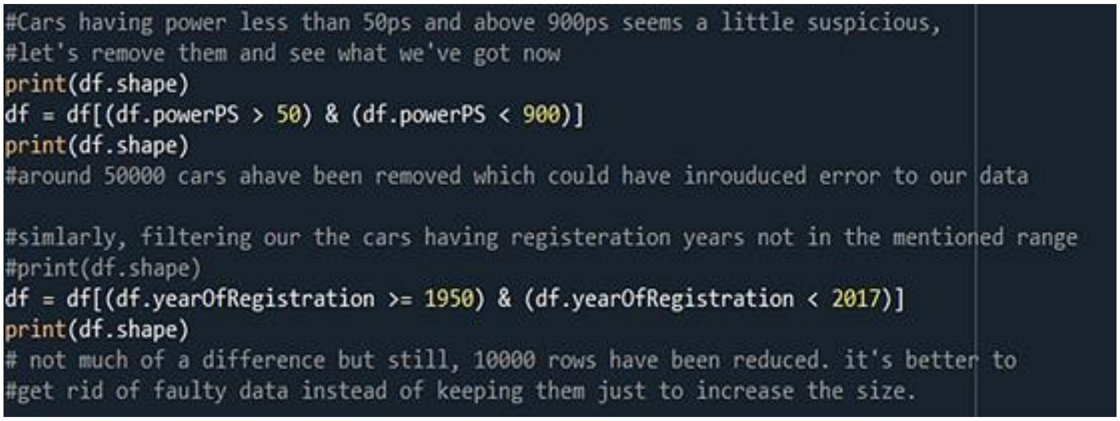
In this activity, the dataset is being cleaned. The dataset contains columns that have almost only one type of data. All the other entries expect the seller (gewerblich) containing only 3 entries are the same hence the column is dropped or removed from the dataset.

Similarly, all the entries except the offer type (Gesuch) containing only 12 entries are same hence offer type column is also dropped.



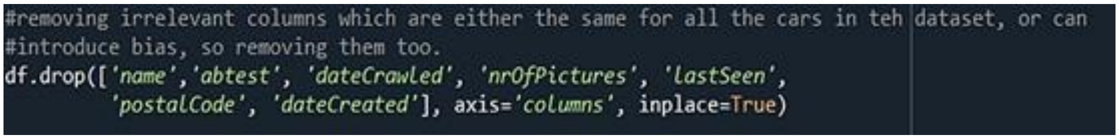
```
#print al the different sellers
print(df.seller.value_counts())
#remove the seller type having only 3 cars
df[df.seller != 'gewerblich']
#now all the sellers are same so we can get rid of this column
df=df.drop('seller',1)

#print al the different sellers
print(df.offerType.value_counts())
#remove the Offer Type having only 12 listings
df[df.offerType != 'Gesuch']
#now all the offers are same so we can get rid of this column
df=df.drop('offerType',1)
```

Suspicious data and data that is not in the range specified are not considered by only taking that is found to be valid. Like the Power should be between 50ps to 900ps and Year of Registration between 1950 and 2017. All the remaining entries are neglected.



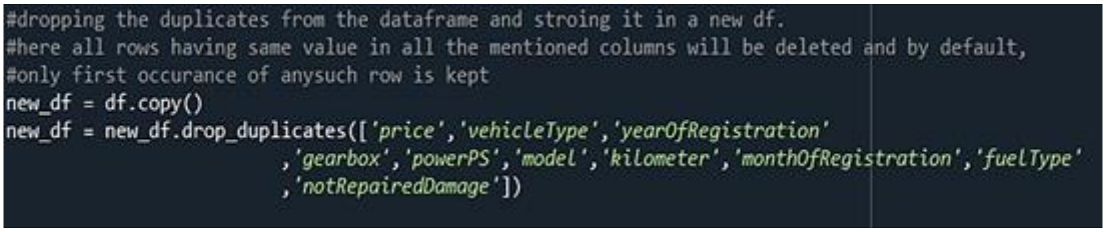
```
#Cars having power less than 50ps and above 900ps seems a little suspicious,
#let's remove them and see what we've got now
print(df.shape)
df = df[(df.powerPS > 50) & (df.powerPS < 900)]
print(df.shape)
#around 50000 cars ahave been removed which could have inrouduced error to our data

#simlarly, filtering our the cars having registeration years not in the mentioned range
#print(df.shape)
df = df[(df.yearOfRegistration >= 1950) & (df.yearOfRegistration < 2017)]
print(df.shape)
# not much of a difference but still, 10000 rows have been reduced. it's better to
#get rid of faulty data instead of keeping them just to increase the size.
```

Columns that introduce bias or are the same for all the cars are removed.



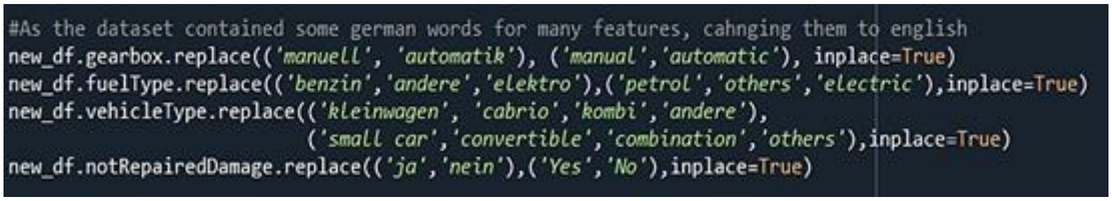
```
#removing irrelevant columns which are either the same for all the cars in teh dataset, or can
#introduce bias, so removing them too.
df.drop(['name','abtest', 'dateCrawled', 'nrOfPictures', 'lastSeen',
        'postalCode', 'dateCreated'], axis='columns', inplace=True)
```

A copy of the dataset is made by deleting the rows that have the same values across all the columns. The first of such rows are stored and the remaining are omitted.



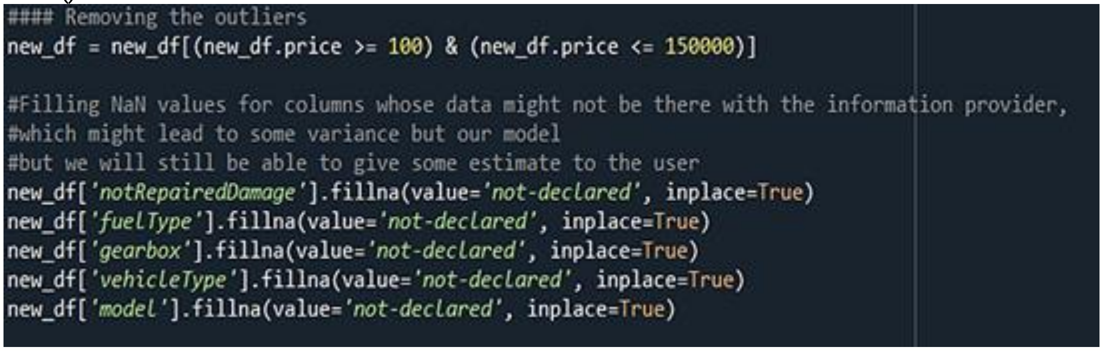
```
#dropping the duplicates from the dataframe and stroing it in a new df.
#here all rows having same value in all the mentioned columns will be deleted and by default,
#only first occurance of anysuch row is kept
new_df = df.copy()
new_df = new_df.drop_duplicates(['price','vehicleType','yearOfRegistration'
                        ,'gearbox','powerPS','model','kilometer','monthOfRegistration','fuelType'
                        ,'notRepairedDamage'])
```

The dataset contains a few German words. Hence replacing the German words with English words.



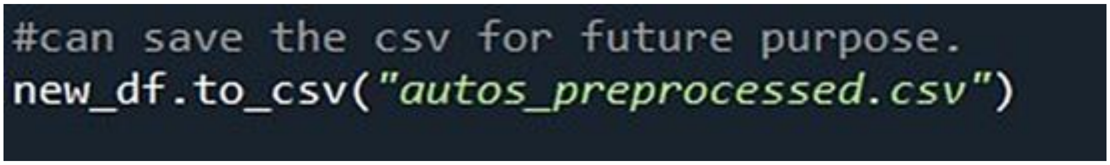
```
#As the dataset contained some german words for many features, cahnging them to english
new_df.gearbox.replace(('manuell', 'automatik'), ('manual','automatic'), inplace=True)
new_df.fuelType.replace(('benzin','andere','elektro'),('petrol','others','electric'),inplace=True)
new_df.vehicleType.replace(('kleinwagen', 'cabrio','kombi','andere'),
                           ('small car','convertible','combination','others'),inplace=True)
new_df.notRepairedDamage.replace(('ja','nein'),('Yes','No'),inplace=True)
```

Removing the outliers (the objects that deviate from the rest of the objects. They can be caused by measurement or execution error) and filling NaN values using the fillna() function.



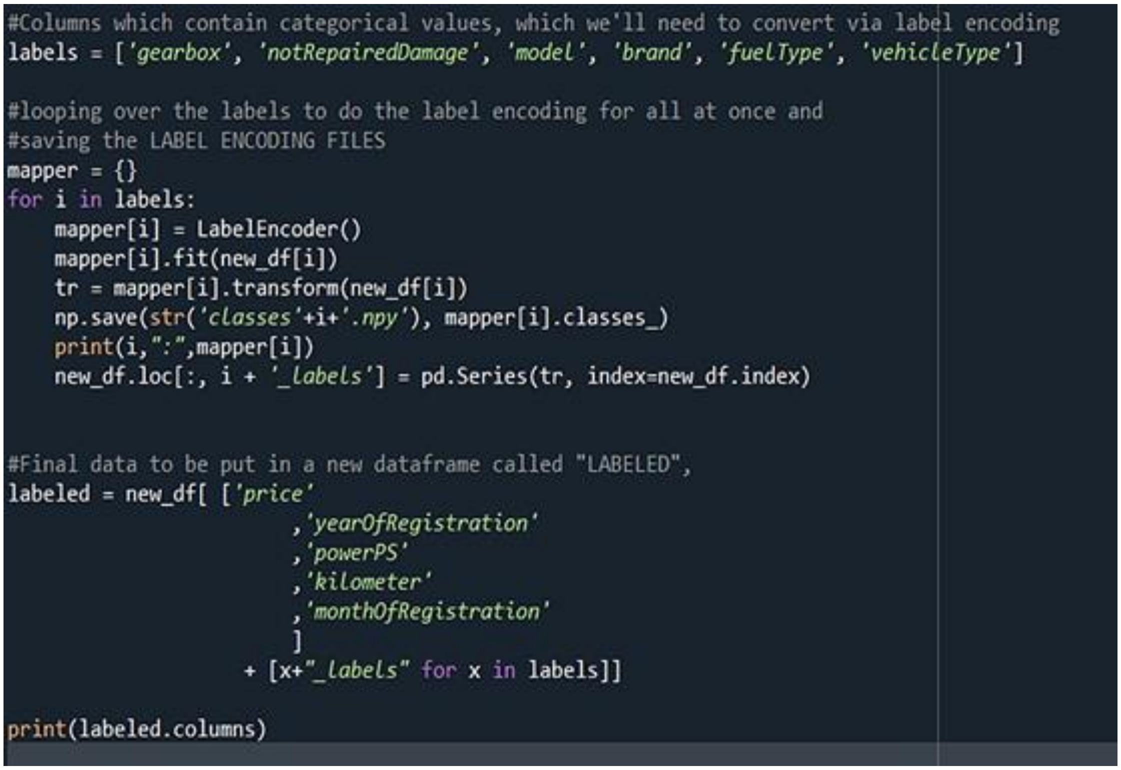
```
#### Removing the outliers
new_df = new_df[(new_df.price >= 100) & (new_df.price <= 150000)]

#Filling NaN values for columns whose data might not be there with the information provider,
#which might lead to some variance but our model
#but we will still be able to give some estimate to the user
new_df['notRepairedDamage'].fillna(value='not-declared', inplace=True)
new_df['fuelType'].fillna(value='not-declared', inplace=True)
new_df['gearbox'].fillna(value='not-declared', inplace=True)
new_df['vehicleType'].fillna(value='not-declared', inplace=True)
new_df['model'].fillna(value='not-declared', inplace=True)
```

Saving the cleaned dataset for future use.



```
#can save the csv for future purpose.
new_df.to_csv("autos_preprocessed.csv")
```

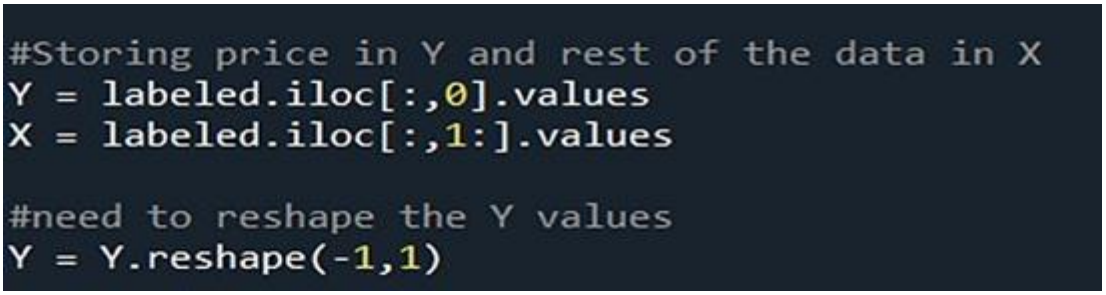
Lable encoding the categirical data.



```
#Columns which contain categorical values, which we'll need to convert via label encoding
labels = ['gearbox', 'notRepairedDamage', 'model', 'brand', 'fuelType', 'vehicleType']

#looping over the labels to do the label encoding for all at once and
#saving the LABEL ENCODING FILES
mapper = {}
for i in labels:
    mapper[i] = LabelEncoder()
    mapper[i].fit(new_df[i])
    tr = mapper[i].transform(new_df[i])
    np.save(str('classes'+i+'.npy'), mapper[i].classes_)
    print(i,":",mapper[i])
    new_df.loc[:, i + '_labels'] = pd.Series(tr, index=new_df.index)


#Final data to be put in a new dataframe called "LABELED",
labeled = new_df[ ['price'
                    ,'yearOfRegistration'
                    ,'powerPS'
                    ,'kilometer'
                    ,'monthOfRegistration'
                    ]
                + [x+"_labels" for x in labels]]

print(labeled.columns)
```

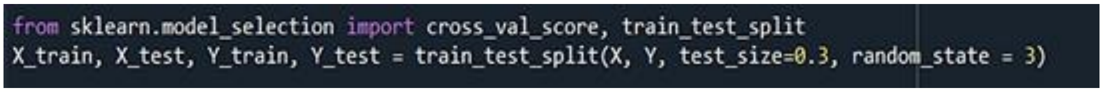
Splitting Data Into Independent And Dependent Variables

In this activity, the dependent and independent variables are to be identified. The first column (Result) in the cleaned dataset is the dependent variable which is dependent on the remaining different factors. The independent columns are considered as x and the dependent column as y.



```
#Storing price in Y and rest of the data in X
Y = labeled.iloc[:,0].values
X = labeled.iloc[:,1:].values

#need to reshape the Y values
Y = Y.reshape(-1,1)
```

After identifying the dependent and independent variables, the dataset now has to be split into two sets, one set is used for training the model and the second set is used for testing how good the model is built. The split ratio we consider is 70% for training and 30% for testing.



```
from sklearn.model_selection import cross_val_score, train_test_split
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.3, random_state = 3)
```

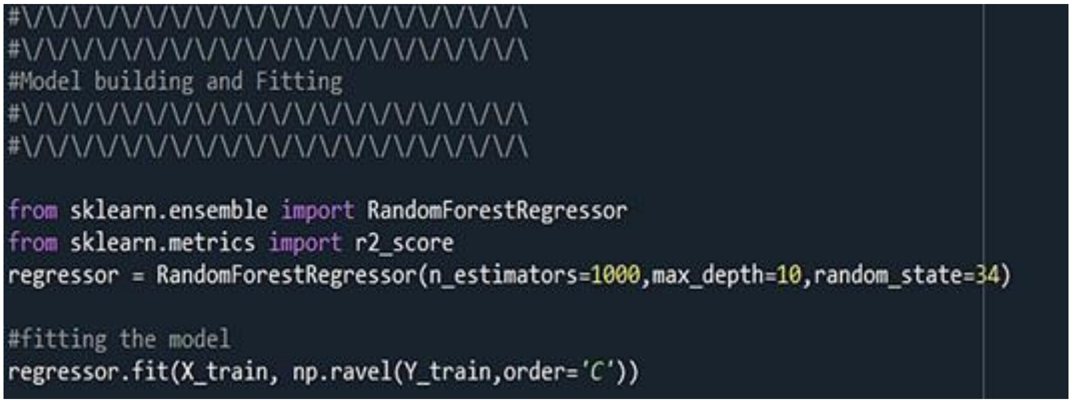
7.2 Model Building

There are several Machine learning algorithms to be used depending on the data you are going to process such as images, sound, text, and numerical values. The algorithms can be chosen according to the objective. As the dataset which we are using is a REgression dataset so you can use the following algorithms.

- Multi Linear Regression
- Random Forest Regression / Classification
- Decision Tree Regression / Classification
- K-Nearest Neighbors
- Support Vector Machine

Choose The Appropriate Model

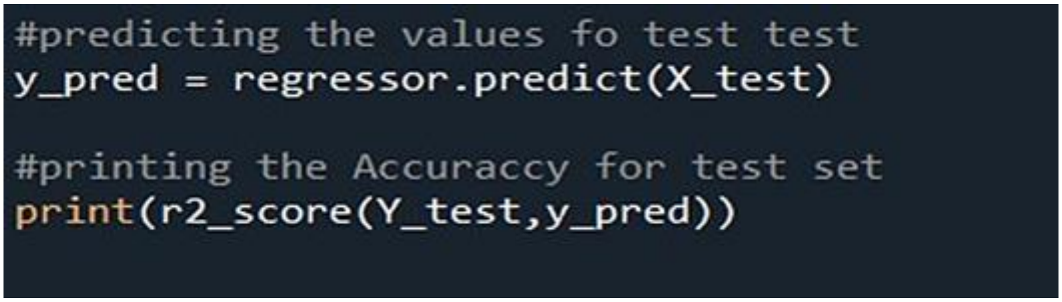
We will be initially considering the Random Forest Regressor model and fit the data.



```
#\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\
#\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\
#Model building and Fitting
#\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\
#\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\

from sklearn.ensemble import RandomForestRegressor
from sklearn.metrics import r2_score
regressor = RandomForestRegressor(n_estimators=1000,max_depth=10,random_state=34)

#fitting the model
regressor.fit(X_train, np.ravel(Y_train,order='C'))
```

Check The Metrics Of The Model

Here we will be evaluating the model built. We will be using the test set for evaluation. The test set is given to the model for prediction and prediction values are stored in another variable called y_pred. The r2 score of the model is calculated and its performance is estimated.



```
#predicting the values fo test test
y_pred = regressor.predict(X_test)

#printing the Accuraccy for test set
print(r2_score(Y_test,y_pred))
```

Note: Different regression models can be used to know the performance and choose whichever works better.

Save The Model

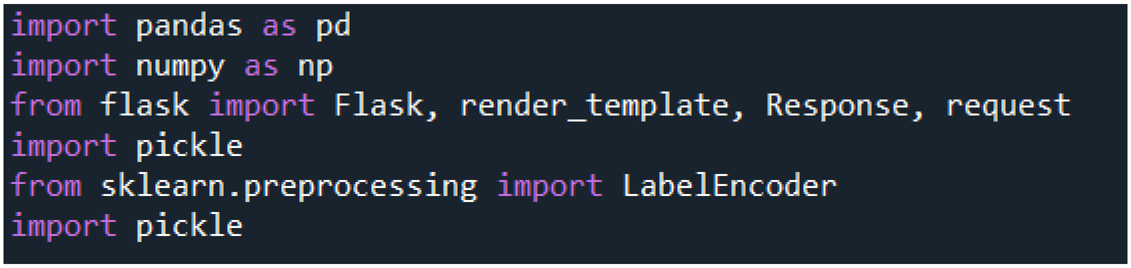
7.3 Application Building

After the model is built, we will be integrating it into a web application so that normal users can also use it to know the resale price of the care. In the application, the user provides the parameter values affecting the resale value.

Build The Python Flask App

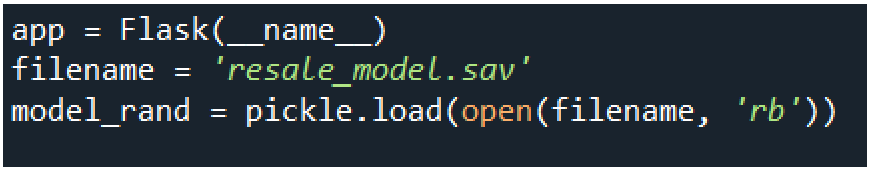
In the flask application, the user values are taken from the HTML page. These factors are then given to the model to know the resale value of a car.

step 1: import required libraries



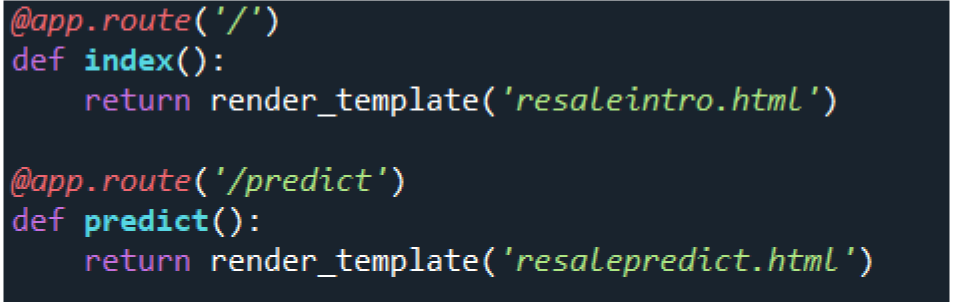
```
import pandas as pd
import numpy as np
from flask import Flask, render_template, Response, request
import pickle
from sklearn.preprocessing import LabelEncoder
import pickle
```

Step 2: Load the model and initialize Flask app

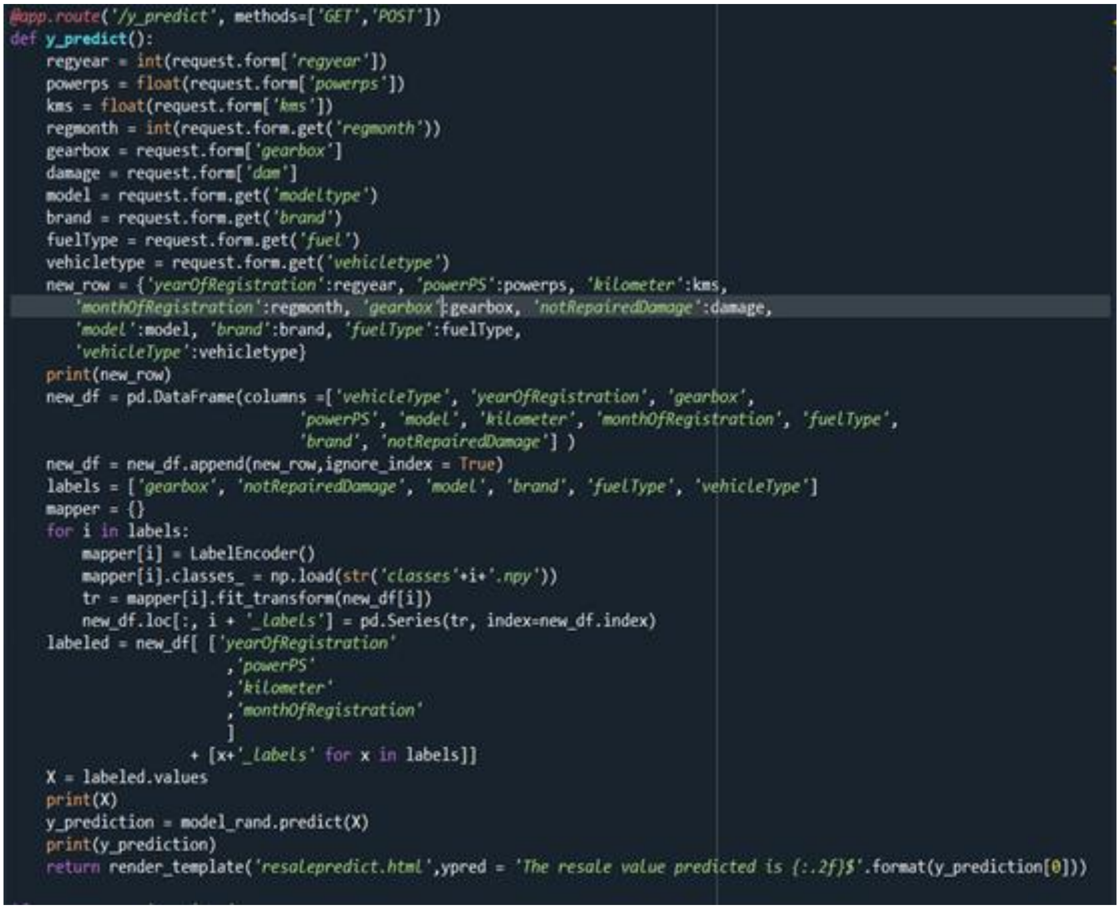


```
app = Flask(__name__)
filename = 'resale_model.sav'
model_rand = pickle.load(open(filename, 'rb'))
```

Step 3: Configure app.py to fetch the parameter values from the UI, and return the prediction.



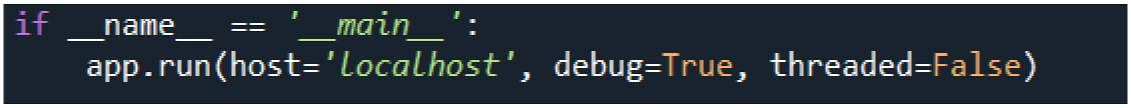
```
@app.route('/')
def index():
    return render_template('resaleintro.html')

@app.route('/predict')
def predict():
    return render_template('resalepredict.html')
```



```python
@app.route('/y_predict', methods=['GET','POST'])
def y_predict():
    regyear = int(request.form['regyear'])
    powerps = float(request.form['powerps'])
    kms = float(request.form['kms'])
    regmonth = int(request.form.get('regmonth'))
    gearbox = request.form['gearbox']
    damage = request.form['dam']
    model = request.form.get('modeltype')
    brand = request.form.get('brand')
    fuelType = request.form.get('fuel')
    vehicletype = request.form.get('vehicletype')
    new_row = {'yearOfRegistration':regyear, 'powerPS':powerps, 'kilometer':kms,
        'monthOfRegistration':regmonth, 'gearbox':gearbox, 'notRepairedDamage':damage,
        'model':model, 'brand':brand, 'fuelType':fuelType,
        'vehicleType':vehicletype}
    print(new_row)
    new_df = pd.DataFrame(columns =['vehicleType', 'yearOfRegistration', 'gearbox',
                            'powerPS', 'model', 'kilometer', 'monthOfRegistration', 'fuelType',
                            'brand', 'notRepairedDamage'] )
    new_df = new_df.append(new_row,ignore_index = True)
    labels = ['gearbox', 'notRepairedDamage', 'model', 'brand', 'fuelType', 'vehicleType']
    mapper = {}
    for i in labels:
        mapper[i] = LabelEncoder()
        mapper[i].classes_ = np.load(str('classes'+i+'.npy'))
        tr = mapper[i].fit_transform(new_df[i])
        new_df.loc[:, i + '_labels'] = pd.Series(tr, index=new_df.index)
    labeled = new_df[ ['yearOfRegistration'
                        ,'powerPS'
                        ,'kilometer'
                        ,'monthOfRegistration'
                        ]
                    + [x+'_labels' for x in labels]]
    X = labeled.values
    print(X)
    y_prediction = model_rand.predict(X)
    print(y_prediction)
    return render_template('resalepredict.html',ypred = 'The resale value predicted is {:.2f}$'.format(y_prediction[0]))
```

Step 4: Run the app

Enter commands as shown below

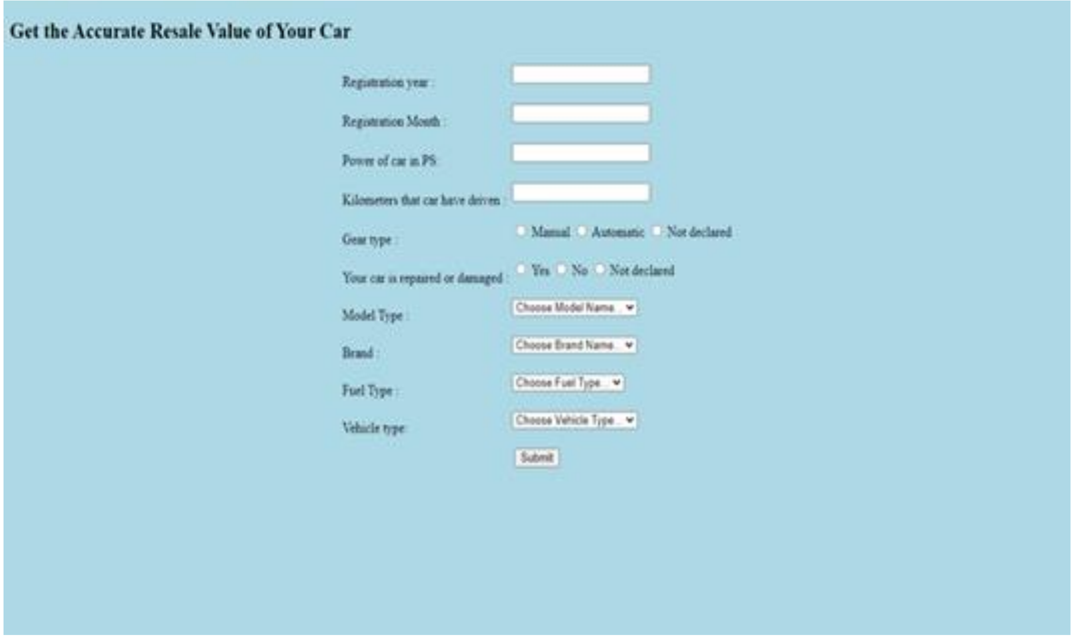


```python
if __name__ == '__main__':
    app.run(host='localhost', debug=True, threaded=False)
```

Build An HTML Page

We Build an HTML page to take the values from the user in a form and upon clicking on the button for submission it has to redirect to URL for **“y_predict”** which returns the predicted resale value. The output is to be then displayed on the page.

The HTML pages are put under the templates folder and any style sheets if present is put in the static folder.



Get the Accurate Resale Value of Your Car

Registration year :

Registration Month :

Power of car in PS:

Kilometers that car have driven :

Gear type : ○ Manual ○ Automatic ○ Not declared

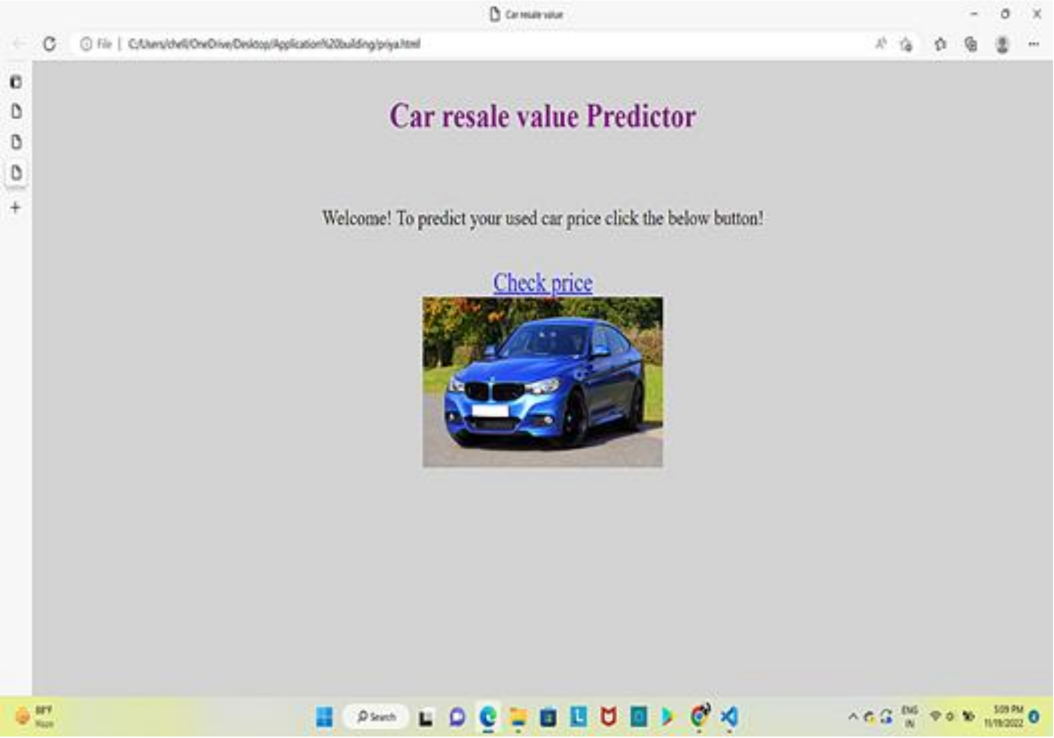
Your car is repaired or damaged : ○ Yes ○ No ○ Not declared

Model Type : Choose Model Name...

Brand : Choose Brand Name...

Fuel Type : Choose Fuel Type...

Vehicle type: Choose Vehicle Type...



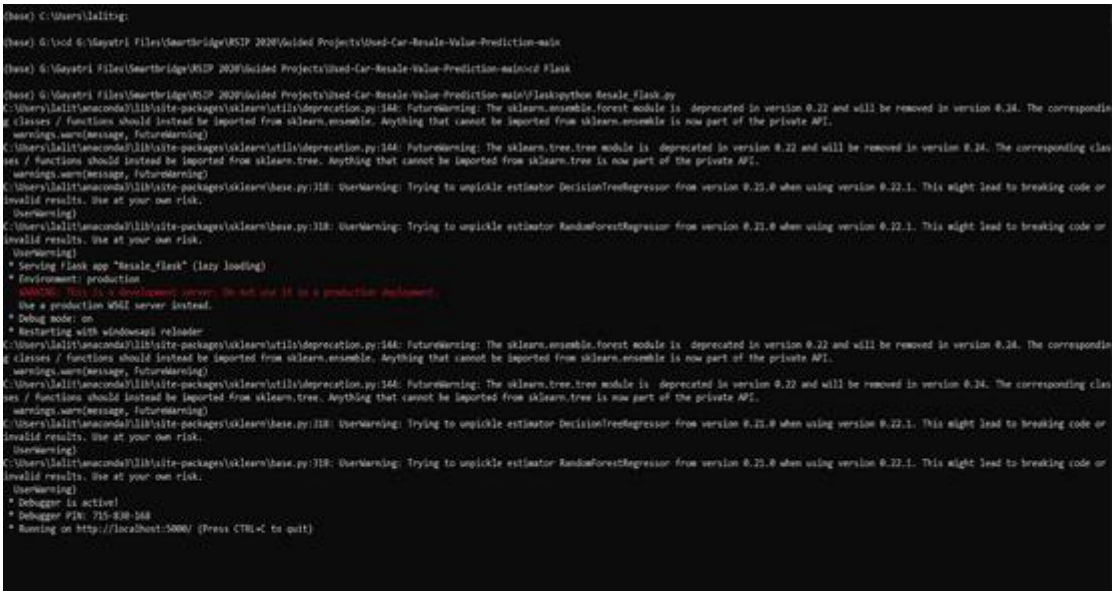
Car resale value
File | C:/Users/chell/OneDrive/Desktop/Application%20building/priya.html
Check price
Search

Car resale value Predictor

Welcome! To predict your used car price click the below button!

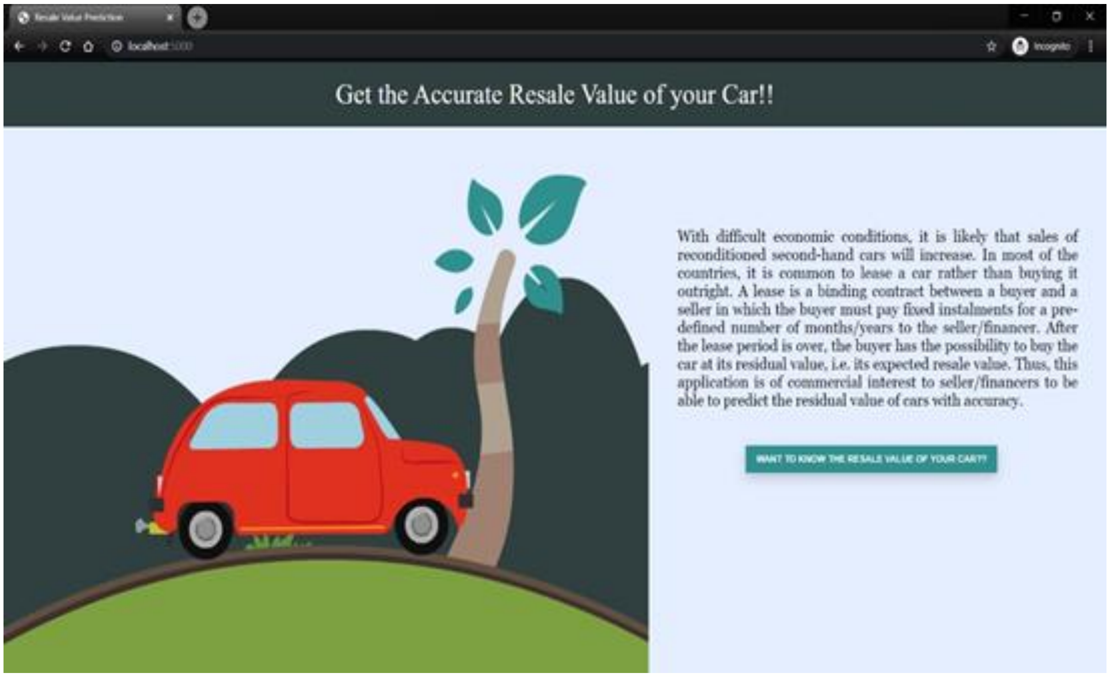
Execute And Test Your Model

Step 1: Execute the python code and after the module is running, open the localhost and click on the button to check the prediction.



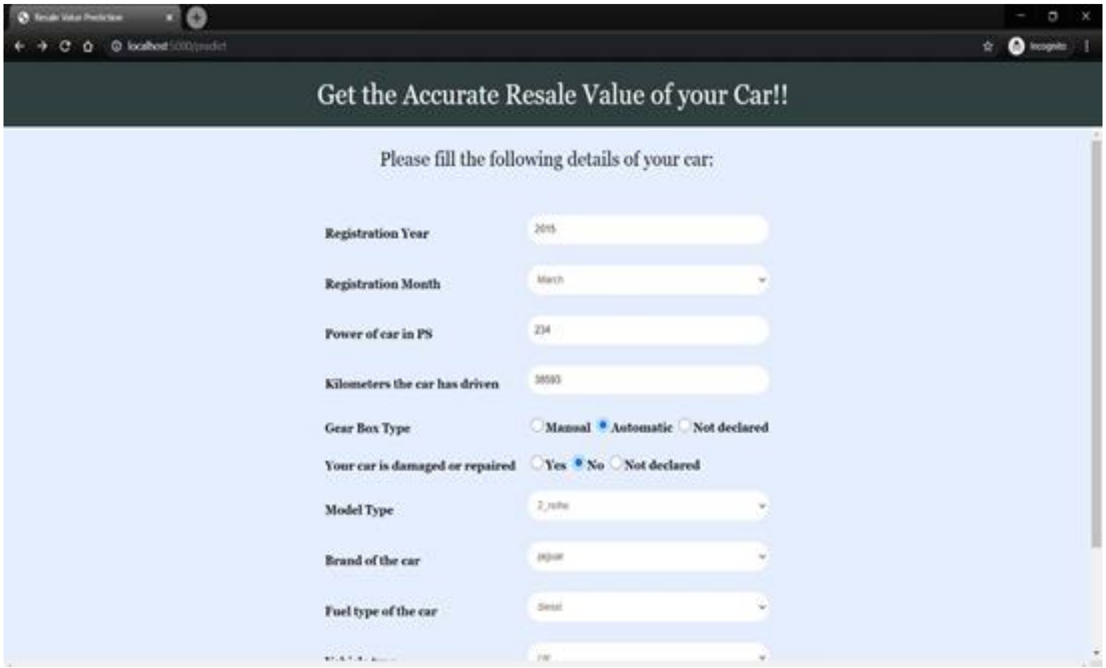
```
(base) C:\Users\lalit>g:

(base) G:\>cd G:\Gayatri Files\Smartbridge\RSIP 2020\Guided Projects\Used-Car-Resale-Value-Prediction-main

(base) G:\Gayatri Files\Smartbridge\RSIP 2020\Guided Projects\Used-Car-Resale-Value-Prediction-main>cd Flask

(base) G:\Gayatri Files\Smartbridge\RSIP 2020\Guided Projects\Used-Car-Resale-Value-Prediction-main\Flask>python Resale_Flask.py
C:\Users\lalit\anaconda3\lib\site-packages\sklearn\utils\deprecation.py:144: FutureWarning: The sklearn.ensemble.forest module is  deprecated in version 0.22 and will be removed in version 0.24. The correspondin
g classes / functions should instead be imported from sklearn.ensemble. Anything that cannot be imported from sklearn.ensemble is now part of the private API.
  warnings.warn(message, FutureWarning)
C:\Users\lalit\anaconda3\lib\site-packages\sklearn\utils\deprecation.py:144: FutureWarning: The sklearn.tree.tree module is  deprecated in version 0.22 and will be removed in version 0.24. The corresponding clas
ses / functions should instead be imported from sklearn.tree. Anything that cannot be imported from sklearn.tree is now part of the private API.
  warnings.warn(message, FutureWarning)
C:\Users\lalit\anaconda3\lib\site-packages\sklearn\base.py:318: UserWarning: Trying to unpickle estimator DecisionTreeRegressor from version 0.21.0 when using version 0.22.1. This might lead to breaking code or
invalid results. Use at your own risk.
  UserWarning)
C:\Users\lalit\anaconda3\lib\site-packages\sklearn\base.py:318: UserWarning: Trying to unpickle estimator RandomForestRegressor from version 0.21.0 when using version 0.22.1. This might lead to breaking code or
invalid results. Use at your own risk.
  UserWarning)
 * Serving Flask app "Resale_flask" (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: on
 * Restarting with windowsapi reloader
C:\Users\lalit\anaconda3\lib\site-packages\sklearn\utils\deprecation.py:144: FutureWarning: The sklearn.ensemble.forest module is  deprecated in version 0.22 and will be removed in version 0.24. The correspondin
g classes / functions should instead be imported from sklearn.ensemble. Anything that cannot be imported from sklearn.ensemble is now part of the private API.
  warnings.warn(message, FutureWarning)
C:\Users\lalit\anaconda3\lib\site-packages\sklearn\utils\deprecation.py:144: FutureWarning: The sklearn.tree.tree module is  deprecated in version 0.22 and will be removed in version 0.24. The corresponding clas
ses / functions should instead be imported from sklearn.tree. Anything that cannot be imported from sklearn.tree is now part of the private API.
  warnings.warn(message, FutureWarning)
C:\Users\lalit\anaconda3\lib\site-packages\sklearn\base.py:318: UserWarning: Trying to unpickle estimator DecisionTreeRegressor from version 0.21.0 when using version 0.22.1. This might lead to breaking code or
invalid results. Use at your own risk.
  UserWarning)
C:\Users\lalit\anaconda3\lib\site-packages\sklearn\base.py:318: UserWarning: Trying to unpickle estimator RandomForestRegressor from version 0.21.0 when using version 0.22.1. This might lead to breaking code or
invalid results. Use at your own risk.
  UserWarning)
 * Debugger is active!
 * Debugger PIN: 715-830-168
 * Running on http://localhost:5000/ (Press CTRL+C to quit)
```

This is the home page of the application



Get the Accurate Resale Value of your Car!!

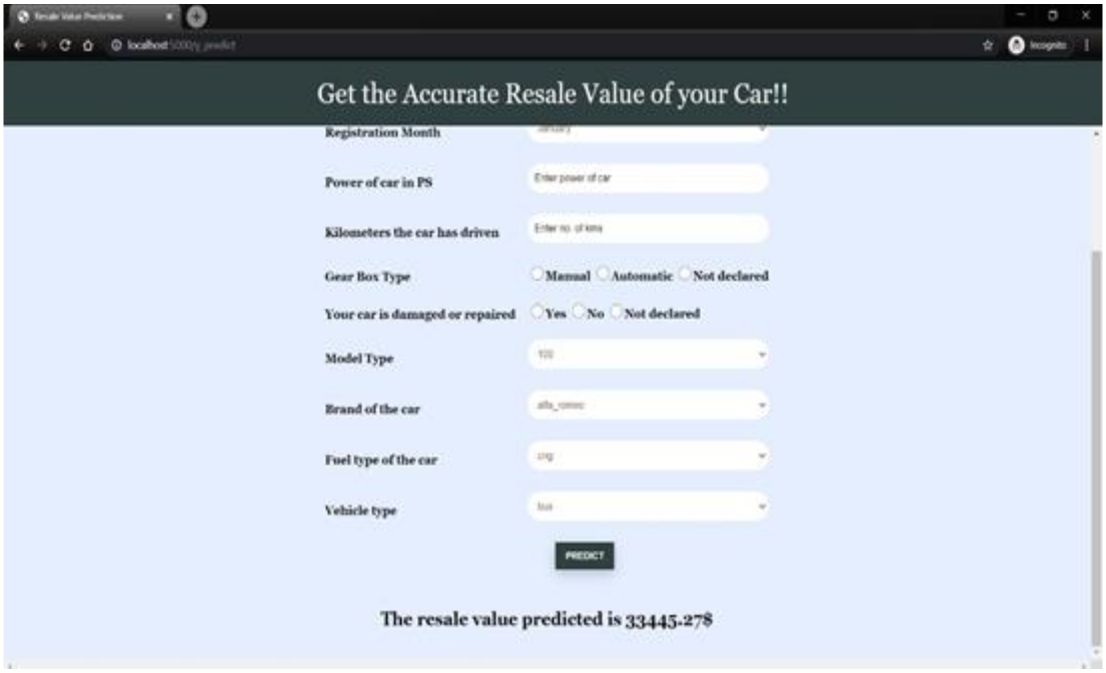
With difficult economic conditions, it is likely that sales of reconditioned second-hand cars will increase. In most of the countries, it is common to lease a car rather than buying it outright. A lease is a binding contract between a buyer and a seller in which the buyer must pay fixed instalments for a pre-defined number of months/years to the seller/financer. After the lease period is over, the buyer has the possibility to buy the car at its residual value, i.e. its expected resale value. Thus, this application is of commercial interest to seller/financers to be able to predict the residual value of cars with accuracy.



Get the Accurate Resale Value of your Car!!

Please fill the following details of your car:

Registration Year	2015
Registration Month	March
Power of car in PS	234
Kilometers the car has driven	38593
Gear Box Type	○ Manual ● Automatic ○ Not declared
Your car is damaged or repaired	○ Yes ● No ○ Not declared
Model Type	2_reihe
Brand of the car	jaguar
Fuel type of the car	diesel
Vehicle type	car



Get the Accurate Resale Value of your Car!!

Registration Month	January
Power of car in PS	Enter power of car
Kilometers the car has driven	Enter no. of kms
Gear Box Type	○ Manual ○ Automatic ○ Not declared
Your car is damaged or repaired	○ Yes ○ No ○ Not declared
Model Type	100
Brand of the car	alfa_romeo
Fuel type of the car	cng
Vehicle type	bus

The resale value predicted is 33445.27$

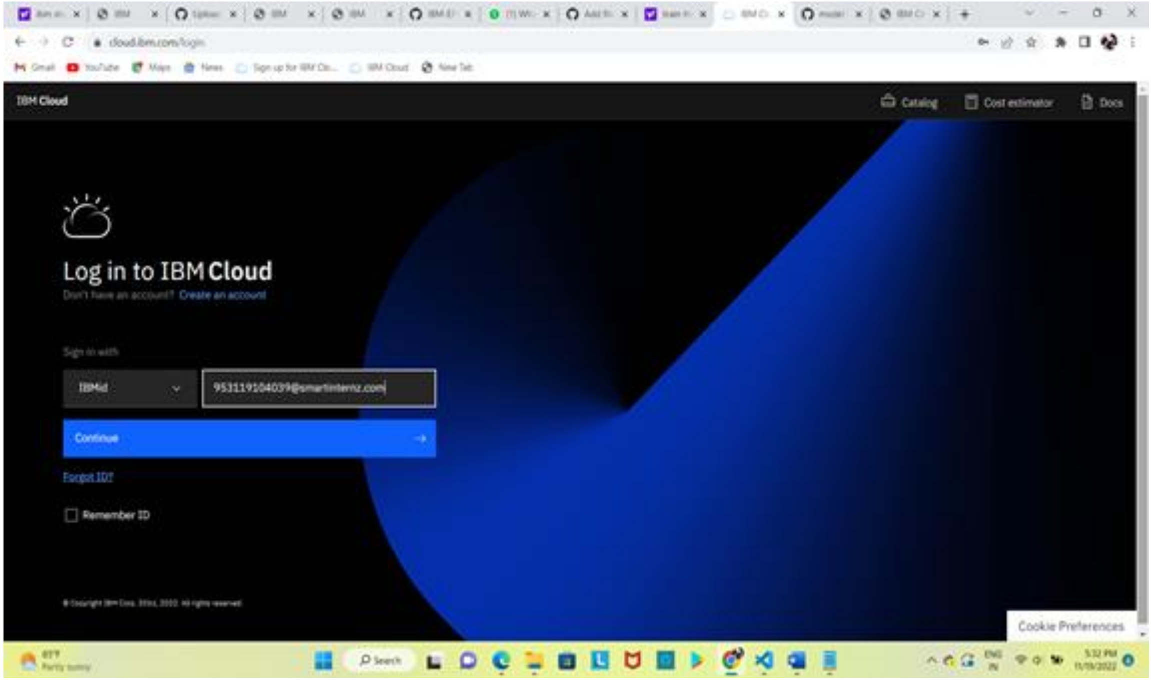
7.4 Train The Model On IBM

In this milestone, you will learn how to build a Machine Learning Model and deploy it on the IBM Cloud.

Register For IBM Cloud

IBM Account:

- Please click here to register for IBM
- Please click here to log in to IBM Account



cloud.ibm.com/login

IBM Cloud

Log in to IBM Cloud

Don't have an account? Create an account

Sign in with

IBMid | 953119104039@smartinternz.com

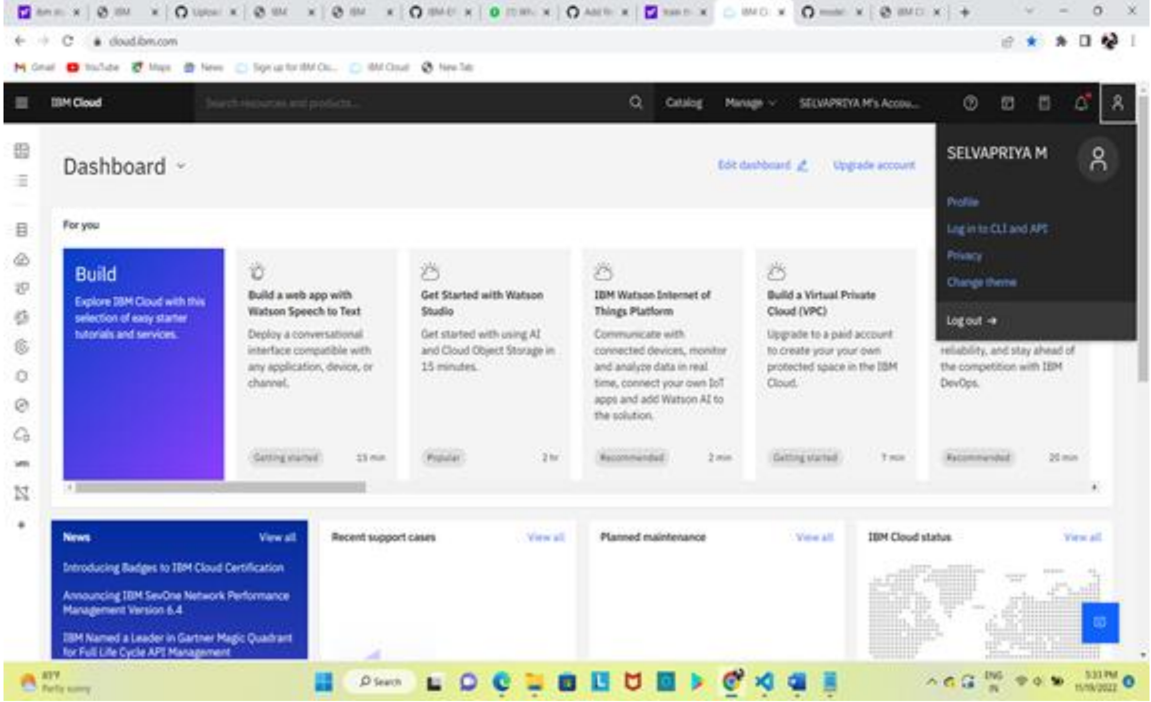
Continue →

Forgot ID?

□ Remember ID

© Copyright IBM Corp. 2014, 2022. All rights reserved.

Cookie Preferences



cloud.ibm.com

IBM Cloud | Catalog | Manage | SELVAPRIYA M's Accou...

Dashboard

Edit dashboard | Upgrade account

SELVAPRIYA M

Profile

Log in to CLI and API

Privacy

Change theme

Log out →

For you

Build: Explore IBM Cloud with this selection of easy starter tutorials and services.

Build a web app with Watson Speech to Text: Deploy a conversational interface compatible with any application, device, or channel. (Getting started, 15 min)

Get Started with Watson Studio: Get started with using AI and Cloud Object Storage in 15 minutes. (Popular, 2 hr)

IBM Watson Internet of Things Platform: Communicate with connected devices, monitor and analyze data in real time, connect your own IoT apps and add Watson AI to the solution. (Recommended, 2 min)

Build a Virtual Private Cloud (VPC): Upgrade to a paid account to create your your own protected space in the IBM Cloud. (Getting started, 7 min)

...reliability, and stay ahead of the competition with IBM DevOps. (Recommended, 20 min)

News — View all

Introducing Badges to IBM Cloud Certification

Announcing IBM SevOne Network Performance Management Version 6.4

IBM Named a Leader in Gartner Magic Quadrant for Full Life Cycle API Management

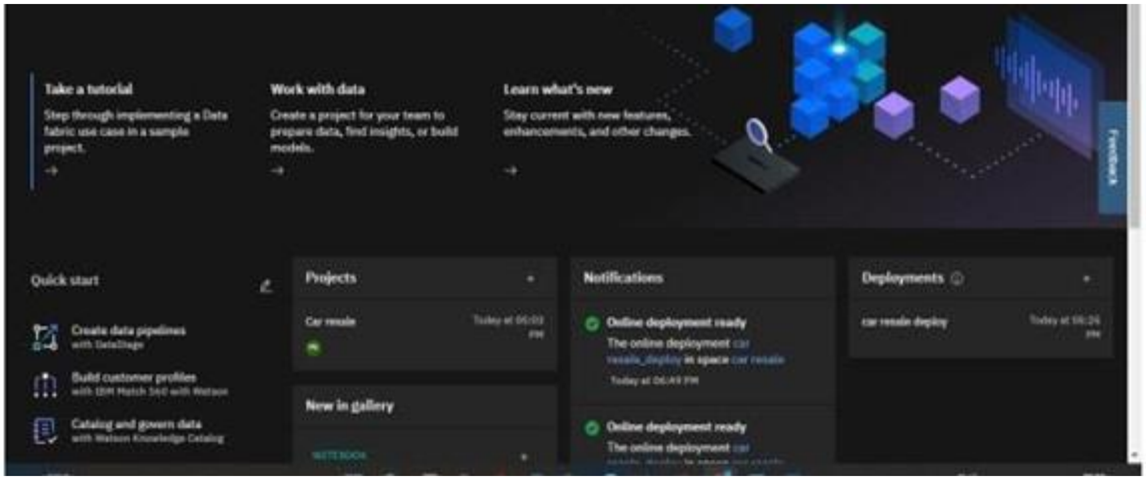
Recent support cases — View all

Planned maintenance — View all

IBM Cloud status — View all

Train The ML Model On IBM

Watch the below video to train the Machine learning model on IBM Watson



Take a tutorial
Step through implementing a Data fabric use case in a sample project.

Work with data
Create a project for your team to prepare data, find insights, or build models.

Learn what's new
Stay current with new features, enhancements, and other changes.

Quick start
- Create data pipelines with DataStage
- Build customer profiles with IBM Match 360 with Watson
- Catalog and govern data with Watson Knowledge Catalog

Projects
Car resale — Today at 06:03 PM

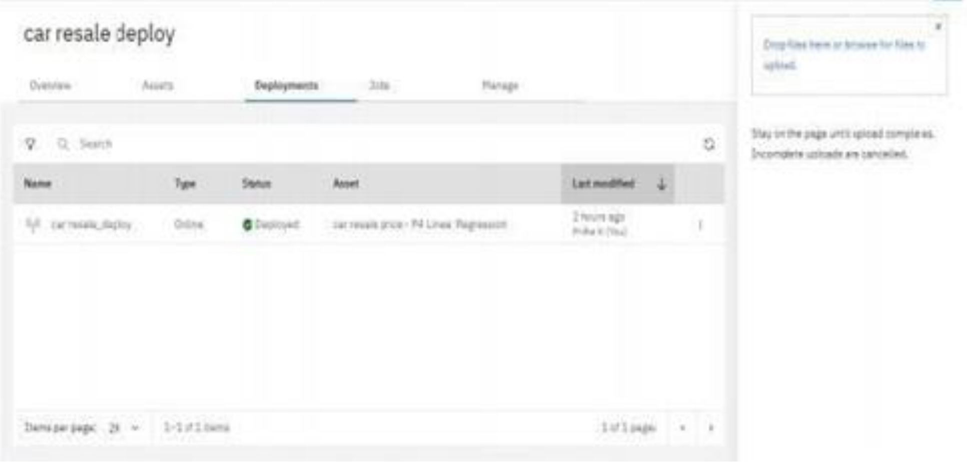
New in gallery
NOTEBOOK

Notifications
Online deployment ready The online deployment car resale_deploy in space car resale — Today at 06:49 PM
Online deployment ready The online deployment car

Deployments
car resale deploy — Today at 06:26 PM

Integrate Flask With Scoring End Point

Watch the below video to integrate the scoring endpoint to the flask



car resale deploy

Overview | Assets | Deployments | Jobs | Manage

Search

Name	Type	Status	Asset	Last modified
car resale_deploy	Online	Deployed	car resale price - P4 Linear Regression	2 hours ago Priha K (You)

Items per page: 25 | 1–1 of 1 items | 1 of 1 pages

Drop files here or browse for files to upload.

Stay on the page until upload completes. Incomplete uploads are cancelled.

8. Testing

8.1 Test Cases

- **Missing values**

The trained ML model requires 4 feature inputs for predicting the output. Failing which, the model throws invalid Input error. All the fields in the html form have been marked required using CSS and thus user must input all fields.

Output: User must input all the fields, failing which, form shows warning message "this field needs to be filled". Thus, there can be no errors in model prediction.

- **Invalid Input**

The trained ML model requires only numerical input for all 4 features. Thus, if user uses symbols such as comma while input, model may throw error. To overcome the same, preprocessing script is deployed in backend which removes all unwanted characters like comma, whitespaces etc. so that model gets required input.

Output: Due to python preprocessing script, model will get the desired input and thus will give accurate prediction.

- **Unseen year of purchase**

The model is trained with data from cars purchased since 2011 to 2020. If the user inputs details of car purchased after that i.e., 2021, model may get confused since that data is quite new and unseen to model.

Output: Model has been trained with boosting algorithm and thus it gives quite accurate results with around RMSE 65,000 INR.

8.2 Users Acceptance Testing

Car resale value prediction system is made with the purpose of predicting the correct valiation of used crs that helps users to sell.

9. Results

In our case, R squared is closer to l, which indicates that the model is reliable in predicting the selling price Randao Forest is Known for attaining high....

10. Advantages and Disadvantages

Advantages

- Good at learning complex and non-linear relationships
- Highly explainable and easy to interpret
- Robust to outliers
- no feature scaling is required

Disadvantages

- Consumes more time
- Requires high computational power

11. Conclusion

- We started with understanding the use case of machine learning in the Automotive industry and how machine learning has transformed the driving experience. Further, we build a Random Forest Regression model to predict the resale value of a used car. Finally, we evaluated the performance of the model using the r squared score and residual plot.
- we could have also used simpler regression algorithms like Linear Regression and Lasso Regression. still, we need to make sure there are no outliers in the dataset before implementing them. Pair plots and scatter plots help visualize the outliers.

12. Future Scope

This Project In machine learning model that will be connected with may dataset and with various website which can provide real time data for priceprediction Will Stored in their site or GitHub. Also, we may add big amount of data of car price which can help an improve accuracy of the machine learningmodel. We also trying to develop an android app as user interface for interacting and user-friendly with user. For better performance of the model, we also plan a to use neural network.

13. Appendix

Project Demo Link:

https://drive.google.com/drive/my drive/demo.link/file:///C:/Users/chell/Downloads/screen-capture.webm

.