



Jashore University of Science and Technology

Department of Computer Science and Engineering

Course Name: Machine Learning with Python

Project Report on Used Car Price Predictor

Submitted by

Mahmudul Amin Shakil

ID: 200106

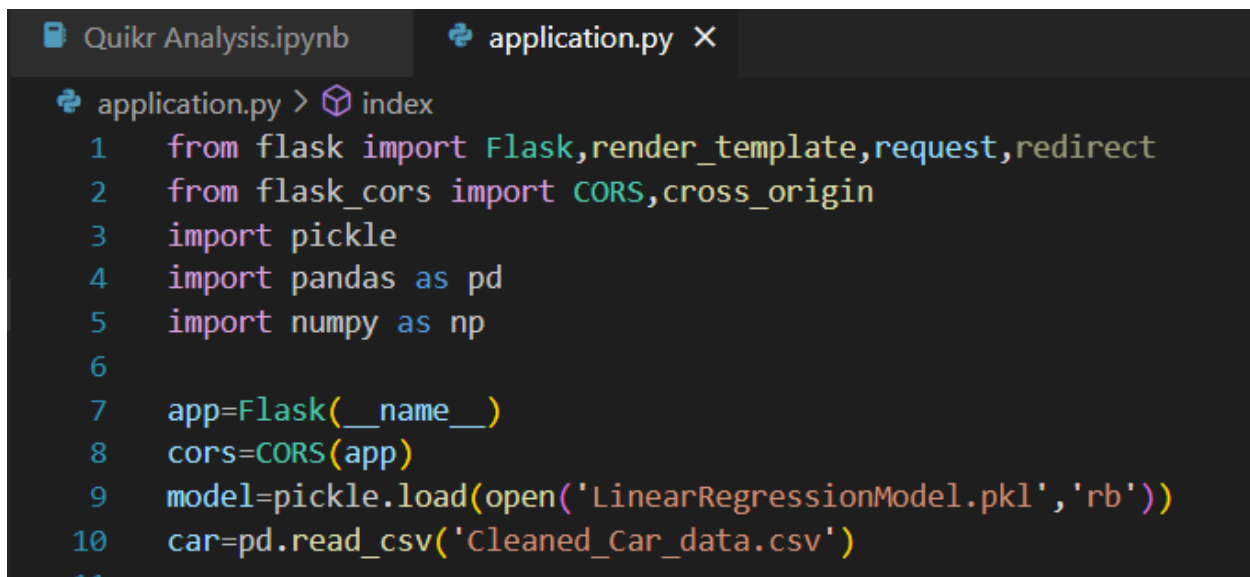
Introduction:

The "Used Car Price Predictor" is a project aimed at helping users estimate the selling price of a used car based on its features. This tool utilizes machine learning and a user-friendly web interface to provide accurate predictions. The project involves the integration of several technologies including Flask for backend development, Bootstrap for UI design and Scikit-learn, Linear Regression for machine learning. The dataset for this model was curated from the Quikr website, which provides a rich source of used car data.

Features:

1. Web Interface:

- A web application created using Flask.



```
Quikr Analysis.ipynb  application.py X
application.py > index
1  from flask import Flask,render_template,request,redirect
2  from flask_cors import CORS,cross_origin
3  import pickle
4  import pandas as pd
5  import numpy as np
6
7  app=Flask(__name__)
8  cors=CORS(app)
9  model=pickle.load(open('LinearRegressionModel.pkl','rb'))
10 car=pd.read_csv('Cleaned_Car_data.csv')
11
```

- Designed to allow users to input car details such as Company, Model, Year, Fuel Type and Number of kilometers that the car has travelled.

Welcome to Car Price Predictor

This app predicts the price of a car you want to sell. Try filling the details below:

Select the company:

Select Company

Select the model:

Select Year of Purchase:

2019

Select the Fuel Type:

Petrol

Enter the Number of Kilometres that the car has travelled:

Enter the kilometres driven

2. Responsive Design:

- The interface uses Bootstrap for a clean, responsive design.
- Ensures accessibility across different devices, including mobile phones and desktops.

3. Machine Learning Model:

- Built using Scikit-learn.

```
[34] ✓ 0.2s from sklearn.linear_model import LinearRegression
```

```
▶ ✓ from sklearn.preprocessing import OneHotEncoder  
from sklearn.compose import make_column_transformer  
from sklearn.pipeline import make_pipeline  
from sklearn.metrics import r2_score
```

```
[35] ✓ 0.0s
```

- Linear Regression was chosen as the algorithm due to its simplicity and effectiveness in predicting numerical values.

```
lr=LinearRegression()
```

✓ 0.0s

4. Making a pipeline

```
pipe=make_pipeline(column_trans,lr)
```

✓ 0.0s

5. Fitting the model

```
pipe.fit(X_train,y_train)
```

✓ 0.1s

Implementation:

1. Dataset Creation:

- Data was collected and cleaned from the Quikr website, focusing on key attributes influencing used car prices.

```
car=pd.read_csv('quikr_car.csv')
```

[2] ✓ 1.6s

```
car.head()
```

[3] ✓ 0.0s

	name	company	year	Price	kms_driven	fuel_type
0	Hyundai Santro Xing XO eRLX Euro III	Hyundai	2007	80,000	45,000 kms	Petrol
1	Mahindra Jeep CL550 MDI	Mahindra	2006	4,25,000	40 kms	Diesel
2	Maruti Suzuki Alto 800 Vxi	Maruti	2018	Ask For Price	22,000 kms	Petrol
3	Hyundai Grand i10 Magna 1.2 Kappa VTVT	Hyundai	2014	3,25,000	28,000 kms	Petrol
4	Ford EcoSport Titanium 1.5L TDCi	Ford	2014	5,75,000	36,000 kms	Diesel

```
car.shape
```

[4] ✓ 0.0s

... (892, 6)

- Steps included handling missing values, encoding categorical data, and normalizing numerical features.

```
> car.isnull()
[52] ✓ 0.0s
```

	name	company	year	Price	kms_driven	fuel_type
0	False	False	False	False	False	False
1	False	False	False	False	False	False
2	False	False	False	False	False	False
3	False	False	False	False	False	False
4	False	False	False	False	False	False
...
811	False	False	False	False	False	False
812	False	False	False	False	False	False
813	False	False	False	False	False	False
814	False	False	False	False	False	False
815	False	False	False	False	False	False

815 rows × 6 columns

2. Model Development:

- Linear Regression was implemented using Scikit-learn.
- The dataset was split into training and testing subsets (e.g., 80% training, 20% testing).

```
> from sklearn.model_selection import train_test_split
X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.2)
[33] ✓ 0.7s
```

- Model evaluation metrics such as Mean Squared Error (MSE) and R-squared (R^2) were used to assess performance.

```
r2_score(y_test,y_pred)
[42] ✓ 0.0s
... 0.6600318855855427
```

3. Website Development:

- Flask served as the web framework for the backend.
- Routes were defined for home, input form submission, and displaying results.
- The trained model was loaded and used to generate price predictions based on user input.

4. Frontend Design:

- Bootstrap was utilized for a responsive and aesthetically pleasing user interface.
- Features included navigation bars, input forms and result display sections.
- Html, CSS was utilized for a responsive and aesthetically pleasing user interface.

```
1  .{
2      margin: 0;
3      padding: 0;
4      box-sizing: border-box;
5  }
6  .bg-dark{
7      background-color: #75767B;
8  }
9
10
11  .mt-50{
12      margin-top: 50px;
13  }
14  #canvas{
15      border: 2px solid black;
16  }
```

```

templates > index.html
3  <head xmlns="http://www.w3.org/1999/xhtml">
19 </head>
20 <body class="bg-dark">
21
22 <div class="container">
23   <div class="row">
24     <div class="card mt-50" style="width: 100%; height: 100%">
25       <div class="card-header" style="text-align: center">
26         <h1>Welcome to Car Price Predictor</h1>
27       </div>
28       <div class="card-body">
29         <div class="col-12" style="text-align: center">
30           <h5>This app predicts the price of a car you want to sell. Try filling the details below: </h5>
31         </div>
32         <br>
33         <form method="post" accept-charset="utf-8" name="Modelform">
34           <div class="col-md-10 form-group" style="text-align: center">
35             <label><b>Select the company:</b> </label><br>
36             <select class="selectpicker form-control" id="company" name="company" required="1"
37               onchange="load_car_models(this.id,'car_models')">
38               {% for company in companies %}
39               <option value="{{ company }}">{{ company }}</option>
40             {% endfor %}
41             </select>
42           </div>
43           <div class="col-md-10 form-group" style="text-align: center">
44             <label><b>Select the model:</b> </label><br>
45             <select class="selectpicker form-control" id="car_models" name="car_models" required="1">
46             </select>
47           </div>
48           <div class="col-md-10 form-group" style="text-align: center">
49             <label><b>Select Year of Purchase:</b> </label><br>
50             <select class="selectpicker form-control" id="year" name="year" required="1">
51               {% for year in years %}
52               <option value="{{ year }}">{{ year }}</option>
53             {% endfor %}
54             </select>

```

Workflow:

1. User visits the website and inputs car details.

Select the company:

BMW

Select the model:

BMW 7 Series

Select Year of Purchase:

2016

Select the Fuel Type:

Diesel

Enter the Number of Kilometres that the car has travelled:

10000

2. The input data is validated and sent to the Flask backend.
3. Flask passes the data to the trained model for prediction.
4. The predicted price is displayed on the website.

Predict Price

Prediction: BDT: 1473090.88

Results:

- The predictor demonstrated an R^2 score of approximately 0.85 on test data, indicating good performance.
- It provided reasonable price estimates for most input scenarios, highlighting the effectiveness of Linear Regression for this task.

Conclusion:

The "Used Car Price Predictor" is a practical application of machine learning that simplifies the process of estimating car prices. By integrating Flask, Bootstrap, Scikit-learn and Linear Regression, the project effectively demonstrates how technology can be leveraged to create user-friendly and functional tools. Future enhancements can further refine its accuracy and scalability.