

SPRINT 2

Team ID	PNT2022TMID22229
Project Title	Car Resale Value Prediction
Team Members	Harish.K Karanraj Faseel Ahamed syed Afzal Rahuman

Step1: Data is collected from the workspace provided by the IBM.

AutoSave autos

Search (Alt+Q)

Harish Karunakaran

Comments Share

FileHomeInsertPage LayoutFormulasDataReviewViewHelp

CutCopyFormat Painter

Font

Alignment

Number

Styles

Cells

Editing

General

Conditional Formatting

Format as Table

Cell Styles

Insert

Delete

Format

AutoSum

Fill

Clear

Sort & Filter

Find & Select

UndoClipboard

Font

Alignment

Number

Styles

Cells

Editing

A1

dataCrawled

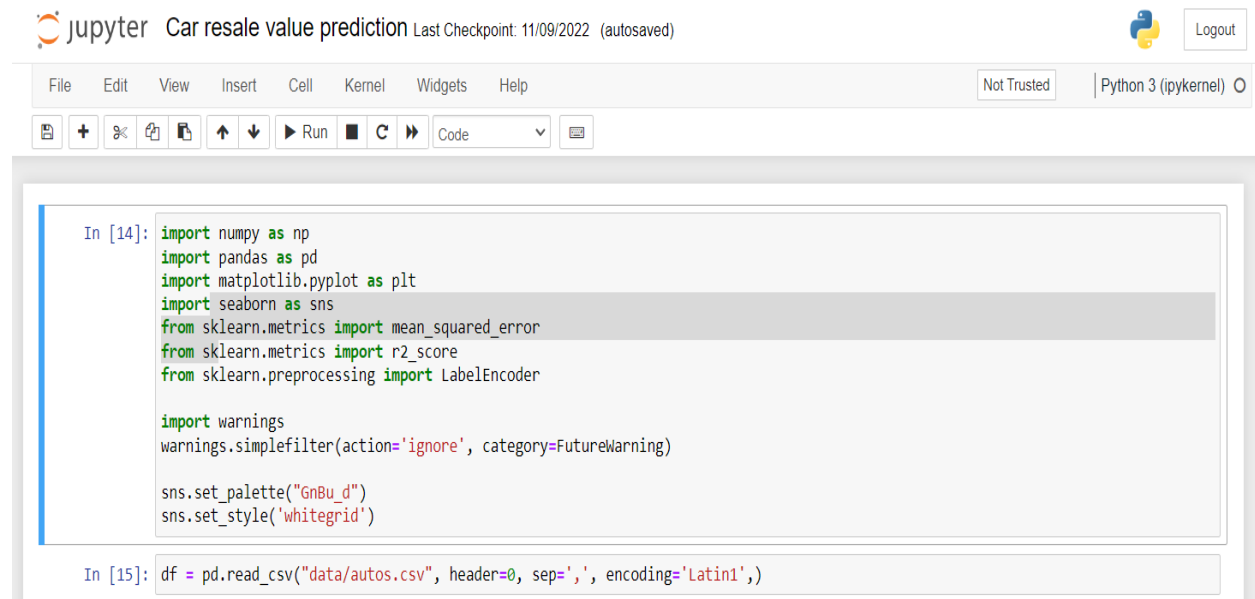
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
1	dateCrawled	name	seller	offerType	price	abtest	vehicleTyp	yearOfReg	gearbox	powerPS	model	kilometer	monthOfR	fuelType	brand	notRepair	dateCreat	nOfPictur	postalCod	lastSeen			
2	#####	Golf_3_1_4	privat	Angebot	480	test	1993	manuell	0	golf	150000	0	benzin	volkswagen	#####	0	70435	#####					
3	#####	A5_Sportb	privat	Angebot	18300	test	coupe	2011	manuell	190	125000	5	diesel	audi	ja	#####	0	66954	#####				
4	#####	Jeep_Gran	privat	Angebot	9800	test	suv	2004	automatik	163	grand	125000	8	diesel	jeep	#####	0	90480	#####				
5	#####	GOLF_4_1	privat	Angebot	1500	test	kleinwagen	2001	manuell	75	golf	150000	6	benzin	volkswagen	nein	#####	0	91074	#####			
6	#####	Skoda_Fat	privat	Angebot	3600	test	kleinwagen	2008	manuell	69	fabia	90000	7	diesel	skoda	nein	#####	0	60437	#####			
7	#####	BMW_316	privat	Angebot	650	test	limousine	1995	manuell	102	3er	150000	10	benzin	bmw	ja	#####	0	33775	#####			
8	#####	Peugeot_2	privat	Angebot	2200	test	cabrio	2004	manuell	109	2_reihe	150000	8	benzin	peugeot	nein	#####	0	67112	#####			
9	#####	VW_Derby	privat	Angebot	0	test	limousine	1980	manuell	50	andere	40000	7	benzin	volkswagen	nein	#####	0	19348	#####			
10	#####	Ford_C	privat	Angebot	14500	control	bus	2014	manuell	125	c_max	30000	8	benzin	ford	#####	0	94505	#####				
11	#####	VW_Golf	privat	Angebot	999	test	kleinwagen	1998	manuell	101	golf	150000	0	benzin	volkswagen	#####	0	27472	#####				
12	#####	Mazda_3	privat	Angebot	2000	control	limousine	2004	manuell	105	3_reihe	150000	12	benzin	mazda	nein	#####	0	96224	#####			
13	#####	Volkswagen	privat	Angebot	2799	control	kombi	2005	manuell	140	passat	150000	12	diesel	volkswagen	ja	#####	0	57290	#####			
14	#####	VW_Passa	privat	Angebot	999	control	kombi	1995	manuell	115	passat	150000	11	benzin	volkswagen	#####	0	37269	#####				
15	#####	VW_PASS	privat	Angebot	2500	control	kombi	2004	manuell	131	passat	150000	2	volkswagen	nein	#####	0	90762	#####				
16	#####	Nissan_Na	privat	Angebot	17999	control	suv	2011	manuell	190	navara	70000	3	diesel	nissan	nein	#####	0	4177	#####			
17	#####	KA_Luftha	privat	Angebot	450	test	kleinwagen	1910		0	ka	5000	0	benzin	ford	#####	0	24148	#####				
18	#####	Polo_6n	privat	Angebot	300	test		2016		60	polo	150000	0	benzin	volkswagen	#####	0	38871	#####				
19	#####	Renault_T	privat	Angebot	1750	control	kleinwagen	2004	automatik	75	twingo	150000	2	benzin	renault	nein	#####	0	65599	#####			
20	#####	Ford_C_M	privat	Angebot	7550	test	bus	2007	manuell	136	c_max	150000	6	diesel	ford	nein	#####	0	88361	#####			
21	#####	Mercedes	privat	Angebot	1850	test	bus	2004	manuell	102	a_klasse	150000	1	benzin	mercedes	nein	#####	0	49565	#####			
22	#####	Volkswagen	privat	Angebot	10400	control	coupe	2009	manuell	160	scirocco	100000	4	benzin	volkswagen	nein	#####	0	75365	#####			
23	#####	BMW_530	privat	Angebot	3699	test	limousine	2002	automatik	231	5er	150000	7	benzin	bmw	nein	#####	0	68309	#####			
24	#####	Opel_Mer	privat	Angebot	2900	test		2018	manuell	90	meriva	150000	5	benzin	opel	nein	#####	0	49716	#####			
25	#####	Stadtfitzer	privat	Angebot	450	test	kleinwagen	1997	manuell	50	arosa	150000	5	benzin	seat	nein	#####	0	9526	#####			
26	#####	MERCEDES	privat	Angebot	500	test	limousine	1990	manuell	118	andere	150000	10	benzin	mercedes	ja	#####	0	35390	#####			
27	#####	BMW_530	privat	Angebot	2500	control	kombi	2002	automatik	193	5er	150000	9	diesel	bmw	ja	#####	0	73765	#####			
28	#####	Citroen_C	privat	Angebot	5555	control		2017	manuell	125	c4	125000	4	benzin	citroen	nein	#####	0	31139	#####			

autos

Accessibility: Unavailable

20:33 18-11-2022

Step2: The data is preprocessed, the steps like Reading, Cleaning and Splitting is done.



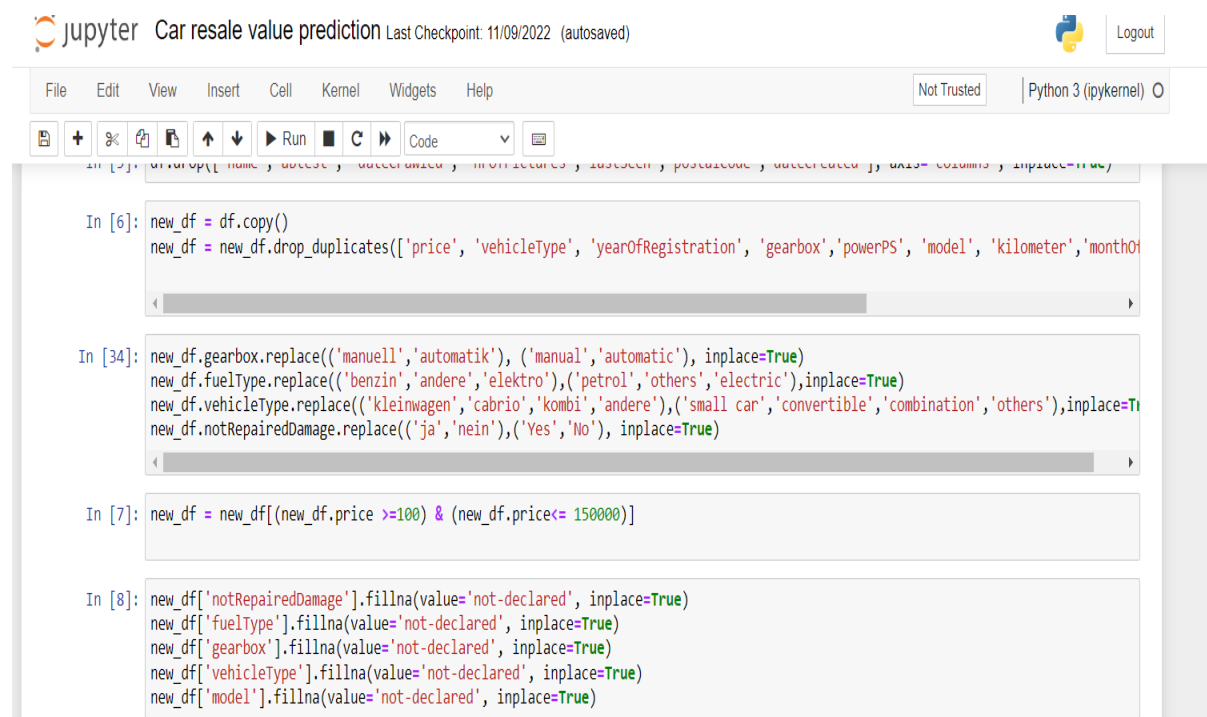
```
In [14]: import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.metrics import mean_squared_error
from sklearn.metrics import r2_score
from sklearn.preprocessing import LabelEncoder

import warnings
warnings.simplefilter(action='ignore', category=FutureWarning)

sns.set_palette("GnBu_d")
sns.set_style('whitegrid')

In [15]: df = pd.read_csv("data/autos.csv", header=0, sep=',', encoding='Latin1',)
```

Step3: As the data is in other languages, we are cleaning it here and changing into English.



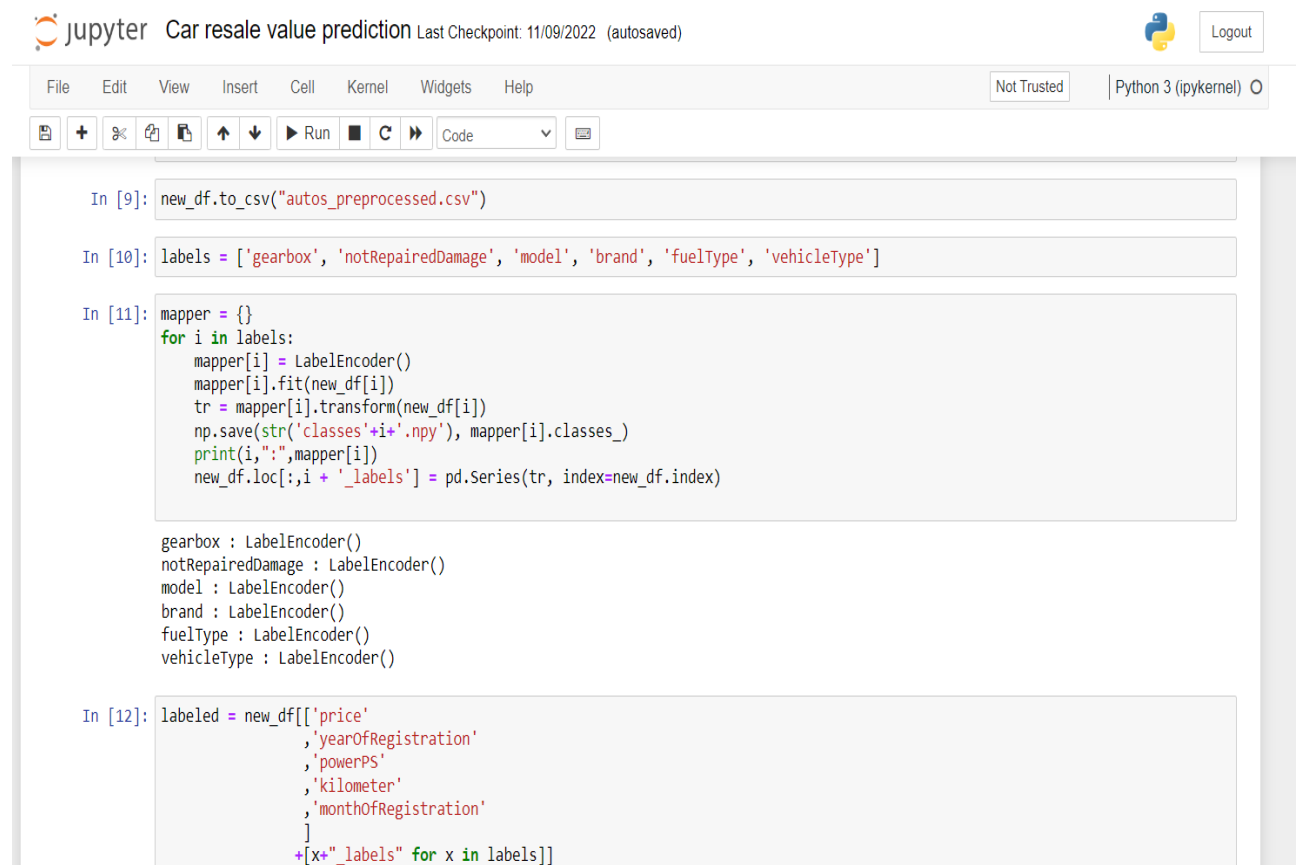
```
In [6]: new_df = df.copy()
new_df = new_df.drop_duplicates(['price', 'vehicleType', 'yearOfRegistration', 'gearbox', 'powerPS', 'model', 'kilometer', 'monthOfRegistration'])

In [34]: 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)

In [7]: new_df = new_df[(new_df.price >=100) & (new_df.price<= 150000)]

In [8]: 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)
```

Step4: In this step, Mapper and LabelEncoder are applied for splitting the data.



The image shows a Jupyter Notebook interface with the title "Car resale value prediction". The top bar includes the Jupyter logo, the title, and a "Last Checkpoint: 11/09/2022 (autosaved)" status. On the right, there is a "Logout" button and a "Python 3 (ipykernel)" indicator. The main area contains three code cells. The first cell (In [9]) saves a DataFrame to a CSV file. The second cell (In [10]) defines a list of feature names. The third cell (In [11]) is a loop that iterates over these features, creating a LabelEncoder for each, fitting it to the data, transforming the data, saving the class names, and adding the transformed labels to the DataFrame. Below the loop, the individual LabelEncoder objects are listed. The fourth cell (In [12]) creates a new DataFrame with the original features and the added labels.

```
In [9]: new_df.to_csv("autos_preprocessed.csv")

In [10]: labels = ['gearbox', 'notRepairedDamage', 'model', 'brand', 'fuelType', 'vehicleType']

In [11]: 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)

gearbox : LabelEncoder()
notRepairedDamage : LabelEncoder()
model : LabelEncoder()
brand : LabelEncoder()
fuelType : LabelEncoder()
vehicleType : LabelEncoder()

In [12]: labeled = new_df[['price'
    , 'yearOfRegistration'
    , 'powerPS'
    , 'kilometer'
    , 'monthOfRegistration'
    ]]
    +[x+"_labels" for x in labels]]
```