## **Cars Dataset**

Make

O Acura

2 Acura

3 Acura

4 Acura

car.shape

(428, 15)

Make

Type

MSRP

Origin

Invoice

DriveTrain

EngineSize

Cylinders

Horsepower

MPG\_Highway Weight

dtype: int64

car.isnull().sum()

MPG\_City

Wheelbase

Length

Make

Model

Type

**MSRP** 

Origin

Invoice

DriveTrain

EngineSize

Cylinders

Horsepower

MPG\_City MPG\_Highway

Weight Wheelbase

Length

In [15]: car.head(5)

Make

O Acura

2 Acura

**3** Acura

4 Acura

Toyota

Ford

BMW

Audi

Honda

Dodge

Volvo

Jaguar

Hyundai Subaru

Pontiac

Mazda

Lexus

Buick

Mercury

Lincoln Saturn

Cadillac

Infiniti

Land Rover

Oldsmobile

Suzuki

GMC Acura Porsche Saab

Jeep

Scion

Isuzu

MINI Hummer

In [17]: car.head(2)

Make

Make

O Acura

1 Acura

2 Acura

**3** Acura

4 Acura

Volvo

Volvo

Volvo

281 rows × 15 columns

**427** Volvo

423

425

426

In [19]: car.head(5)

Make

O Acura

2 Acura

3 Acura

4 Acura

0

15

17

18

20

416

Out[21]:

401 Volkswagen

**411** Volkswagen

**412** Volkswagen

415 Volkswagen

Make

1 Acura

2 Acura

**3** Acura

4 Acura

424

426

**425** Volvo

**427** Volvo

car.head(2)

Make

**0** Acura

In [26]: car.head(10)

Make

0 Acura

1 Acura

2 Acura

3 Acura

4 Acura

**5** Acura

6 Acura

7 Audi

9 Audi

Audi

Aman Choudhary

In [23]:

Out[23]:

Out[26]:

Volvo

325 rows × 15 columns

Out[19]:

Out[20]:

O Acura

Out[17]:

Out[18]:

Kia

Nissan

Volkswagen Chrysler

Mitsubishi

Chevrolet Mercedes-Benz

Out[15]:

In [16]:

Out[16]:

dtype: int64

Model

Out[9]:

In [10]:

Out[10]:

In [11]:

Out[11]:

In [14]:

Out[14]:

Here, The data of different cars is given with their specifications.

import pandas as pd

SUV

Model

MDX

TSX 4dr Sedan

TL 4dr Sedan

3.5 RL 4dr Sedan

1 Acura RSX Type S 2dr Sedan

1) Instruction (for Data Cleaning)

0

0

0

0

0

0

0

0

2

0 0

0

0

0

0

0

0

0

0

0

0

0

0

0

0 0

0 0

0 0

2) Question (Based on Value Counts)

Model

MDX

TSX 4dr Sedan

TL 4dr Sedan

3.5 RL 4dr Sedan

28

27

26 23

20

19

17

17

15

15

13

13

12

12 12

11

11

11 11

11

9

9

9

8

8

8

8

7

3

3

3 2

2

2

MDX

In [18]: car[car["Origin"].isin(["Asia" , "Europe"])]

1 Acura RSX Type S 2dr Sedan

SUV

Model

MDX

RSX Type S 2dr Sedan

TSX 4dr Sedan

TL 4dr Sedan

S80 T6 4dr Sedan Europe

V40 Wagon Europe

XC70 Wagon Europe

Type Origin DriveTrain

Asia

Asia

Asia

Asia

Asia

A6 4.2 Quattro 4dr

A8 L Quattro 4dr

RS 6 4dr

Touareg V6

XC90 T6

Type

Sedan

Sedan

S80 T6 4dr Sedan Europe

V40 Wagon Europe

XC70 Wagon Europe

- Increase all the value of 'MPG\_City' column by 3.

Type Origin DriveTrain

Asia

Asia

Type

SUV

Sedan

car["MPG\_City"] = car["MPG\_City"].apply(lambda x: x+3)

Sedan Europe

Phaeton W12 4dr

Model

RSX Type S 2dr Sedan

TL 4dr

3.5 RL 4dr

**5** Acura 3.5 RL w/Navigation 4dr Sedan

5) Instruction (Applying Fuction on a column)

MDX

1 Acura RSX Type S 2dr Sedan

SUV

Model

MDX

TSX 4dr Sedan

3.5 RL 4dr Sedan

A4 1.8T 4dr Sedan

A4 3.0 4dr Sedan Europe

RSX Type S 2dr Sedan

TL 4dr

3.5 RL w/Navigation 4dr Sedan

NSX coupe 2dr manual S Sports

Email - amanchoudhary11189.ac@gmail.com

linkedin - www.linkedin.com/in/aman-choudhary-61a9361a0

A41.8T convertible 2dr Sedan

Volvo C70 HPT convertible 2dr

423 Volvo C70 LPT convertible 2dr Sedan Europe

TSX 4dr Sedan

Model

MDX

Type

SUV

Sedan Europe

Sedan Europe

Sedan Europe

Sports Europe

Sedan Europe

SUV Europe

Origin DriveTrain

Asia

Asia

Asia

Asia

Asia

Europe

SUV

Phaeton 4dr Sedan Europe

Passat W8 Wagon Europe

- Remove all the records (rows) where weight is above 4000.

All \$36,945 \$33,337

Front \$23,820 \$21,761

Front \$26,990 \$24,647

Front \$33,195 \$30,299

Front \$43,755 \$39,014

Asia

Origin DriveTrain

3.5 RL 4dr Sedan

C70 LPT convertible 2dr Sedan Europe

424 Volvo C70 HPT convertible 2dr Sedan Europe

4) Instruction (Removing Unwanted records)

MDX

TSX 4dr Sedan

TL 4dr Sedan

Audi A4 3.0 Quattro convertible 2dr

3.5 RL 4dr Sedan

1 Acura RSX Type S 2dr Sedan

car[car['Weight'] > 4000]

Make

Acura

Audi

Audi

Audi

Volvo

car[~(car["Weight"] > 4000)]

103 rows × 15 columns

SUV

- Show all the records where origin is Asia or Europe.

Type Origin DriveTrain

Type

SUV

Asia

Asia

Asia

Asia

Asia

Asia

Asia

All \$36.945

Front \$23,820 \$21,761

\$33,337

All \$36,945 \$33,337

Front \$23,820 \$21,761

Front \$26,990 \$24,647

Front \$33,195 \$30,299

Front \$43,755 \$39,014

Front \$40,565 \$38,203

Front \$42,565 \$40,083

Front \$45,210 \$42,573

Front \$26,135 \$24,641

All \$35,145 \$33,112

3.5

2.0

2.4

3.2

3.5

All \$36,945 \$33,337

All \$44,240 \$40,075

All \$49,690 \$44,936

All \$69,190 \$64,740

All \$35,515 \$32,243

Front \$65,000 \$59,912

Front \$75,000 \$69,130

Front \$40,235 \$36,956

Front \$23,820 \$21,761

Front \$26,990 \$24,647

Front \$33,195 \$30,299

Front \$43,755 \$39,014

Front \$46,100 \$41,100

Front \$40,565 \$38,203

Front \$42,565 \$40,083

Front \$45,210 \$42,573

Front \$26,135 \$24,641

All \$36,945 \$33,337

All \$36,945 \$33,337

Front \$23,820 \$21,761

Front \$26,990 \$24,647

Front \$33,195 \$30,299

Front \$43,755 \$39,014

Front \$46,100 \$41,100

Rear \$89,765 \$79,978

Front \$25,940 \$23,508

Front \$35,940 \$32,506

Front \$31,840 \$28,846

Front \$23,820 \$21,761

Origin DriveTrain

Asia

Asia

Asia

Asia

Asia

Asia

Asia

Europe

Europe

All \$35,145 \$33,112

3.5

2.0

All \$41,250 \$38,851

Front \$84,600 \$76,417

Name: Make, dtype: int64

3) Instruction (Filtering)

1 Acura RSX Type S 2dr Sedan

car["Make"].value\_counts()

SUV

car.isnull().sum()

Type Origin DriveTrain

Asia

Asia

Asia

Asia

Asia

car["Cylinders"].fillna(car["Cylinders"].mean(), inplace= True)

Type Origin DriveTrain

Asia

Asia

Asia

Asia

Asia

MSRP

All \$36,945 \$33,337

Front \$23,820 \$21,761

Front \$26,990 \$24,647

Front \$33,195 \$30,299

Front \$43,755 \$39,014

car = pd.read\_csv(r"C:\Users\hp\Downloads\file.csv")

In [9]: car.head()

This data is available as a CSV file. We are going to analyze the data of using the Pandas DataFrame.

All \$36,945 \$33,337

Front \$23,820 \$21,761

Front \$26,990 \$24,647

Front \$33,195 \$30,299

Front \$43,755 \$39,014

3.5

2.0

2.4

3.2

3.5

6.0

4.0

4.0

6.0

6.0

- Find all the Null Value in the dataset, if there is any null value in any cloumn then fit it with the mean of that column.

- Check what are the different type of Make are there in our dataset. And what is the count (occurance) of each Make in the data?

6.0

4.0

4.0

6.0

6.0

3.5

2.0

2.4

3.2

3.5

Invoice EngineSize Cylinders Horsepower MPG\_City MPG\_Highway Weight Wheelbase Length

17

24

22

20

18

265

200

200

270

225

MSRP Invoice EngineSize Cylinders Horsepower MPG\_City MPG\_Highway Weight Wheelbase Length

Origin DriveTrain MSRP Invoice EngineSize Cylinders Horsepower MPG\_City MPG\_Highway Weight Wheelbase Length

17

24

265

200

200

270

225

197

242

268

170

208

23

31

17

24

22

20

18

21

20

19

22

20

4451

2778

106

101

4451

2778

3230

3575

3880

3450

3450

3653

2822

3823

31

29

28

24

28

26

26

29

4451

2778

3230

3575

3880

17

18

17

17

15

15

16

12

18

15

31

29

28

24

24

28

26

26

29

27

2778

3230

3575

3880

3893

3450

3450

3653

2822

3823

106

101

189

172

106

101

105

108

115

115

100

104

105

104

189 172

183

186

197

197

174

179

180

179

23

31

29

28

24

MSRP Invoice EngineSize Cylinders Horsepower MPG\_City MPG\_Highway Weight Wheelbase Length

265

220

300

330

450

220

335

420

270

268

24

22

20

18

18

21

20

19

22

20

23

31

4451

2778

23

31

29

28

24

24

24

31

30

28

4451

2778

3230

3575

3880

3893

3153

3252

3638

3462

189

172

106

101

105

108

115

105

105

110

101

109

189

172

183

186

197

4451

4013

4024

4399

4024

5086

5194

5399

4067

4638

106

105

109

121

109

112

118

118

106

113

Length

172

183

186

197

197

186

186

190

180

186

101

105

108

115

115

105

105

110

101

109

189

180

193

204

191

187

204

204

184

189

106

101

105

108

115

23

25

24

24

22

20

22

19

25

20

189

172

183

186

197

186

186

190

180

186

265

200

6.0

4.0

4.0

6.0

6.0

5.0

5.0

6.0

4.0

5.0

MSRP Invoice EngineSize Cylinders Horsepower MPG\_City MPG\_Highway Weight Wheelbase Length

17

24

22

20

18

6.0

6.0

8.0

8.0

8.0

6.0

8.0

12.0

8.0

6.0

MSRP Invoice EngineSize Cylinders Horsepower MPG\_City MPG\_Highway Weight Wheelbase

200

200

270

225

225

197

242

268

170

208

265

200

200

270

225

3.5

3.0

4.2

4.2

4.2

4.2

6.0

4.0

2.9

4.0

4.0

6.0

6.0

6.0

5.0

5.0

6.0

4.0

5.0

MSRP Invoice EngineSize Cylinders Horsepower MPG\_City MPG\_Highway Weight Wheelbase Length

17

24

MSRP Invoice EngineSize Cylinders Horsepower MPG\_City MPG\_Highway Weight Wheelbase Length

20

27

25

23

21

21

20

25

26

23

265

200

200

270

225

225

290

170

170

220

265

200

6.0

4.0

4.0

6.0

6.0

6.0

6.0

4.0

4.0

6.0

2.0

2.4

3.2

3.5

3.5

2.4

2.3

2.9

1.9

2.5

6.0

4.0

3.5

2.0

2.4

3.2

3.5

3.5

3.2

1.8

1.8

3.0

6.0

4.0

3.5

2.0

2.4

3.2

3.5

2.4

2.3

2.9

1.9

2.5

6.0

4.0

4.0

6.0

6.0

3.5

2.0

4451

2778

3230

3575

3880

23

31

29

28

24

189

172

183

186

197

106

101

105

108

115

MSRP Invoice EngineSize Cylinders Horsepower MPG\_City MPG\_Highway Weight Wheelbase Length

17

24

22

20

18

23

31

29

28

24

4451

2778

3230

3575

3880

106

101

105

108

115

189

172

183

186

197

265

200

200

270

225