

Student Name : Narendra Mamilla

Student ID : 16351892

## Assignment - 2

### 1. Loading the train.csv file into google collab

The screenshot shows the Google Colab interface with a Jupyter Notebook titled 'Untitled0.ipynb'. The file explorer on the left shows a folder named 'data\_raw' containing 'train.csv'. The code cell contains the following Python code:

```
# Import necessary libraries
import pandas as pd
from datetime import datetime

[4] df = pd.read_csv("/content/data_raw/train.csv")
df.head()
```

The output of the code is a preview of the first 5 rows of the 'train.csv' file. The columns are: Unnamed: 0, Name, Location, Year, Kilometers\_Driven, Fuel\_Type, Transmission, Owner\_Type, Mileage, Engine, Power, Seats, New\_Price, and Price. The data rows are as follows:

	Unnamed: 0	Name	Location	Year	Kilometers_Driven	Fuel_Type	Transmission	Owner_Type	Mileage	Engine	Power	Seats	New_Price	Price
0	1	Hyundai Creta 1.6 CRDI SX Option	Pune	2015	41000	Diesel	Manual	First	19.67 kmpl	1582 CC	126.2 bhp	5.0	NaN	12.50
1	2	Honda Jazz V	Chennai	2011	46000	Petrol	Manual	First	13 km/kg	1199 CC	88.7 bhp	5.0	8.61 Lakh	4.50
2	3	Maruti Ertiga VDI	Chennai	2012	87000	Diesel	Manual	First	20.77 kmpl	1248 CC	88.76 bhp	7.0	NaN	6.00
3	4	Audi A4 New 2.0 TDI Multitronic	Coimbatore	2013	40670	Diesel	Automatic	Second	15.2 kmpl	1968 CC	140.8 bhp	5.0	NaN	17.74
4	6	Nissan Micra Diesel XV	Jaipur	2013	86999	Diesel	Manual	First	23.08 kmpl	1461 CC	63.1 bhp	5.0	NaN	3.50

Next steps: [Generate code with df](#) [View recommended plots](#)

### 2. Identifying the missing values

The screenshot shows the Google Colab interface with a Jupyter Notebook. The code cell contains the following Python code:

```
# Identify missing values
missing_values = df.isnull().sum()
print("Missing values in each column:")
print(missing_values)
```

The output of the code is a text representation of the missing values in each column:

```
Missing values in each column:
Unnamed: 0      0
Name            0
Location        0
Year            0
Kilometers_Driven  0
Fuel_Type       0
Transmission     0
Owner_Type      0
Mileage         2
Engine          36
Power           36
Seats           38
New_Price      5032
Price           0
dtype: int64
```

### 3. Removing the units from Power, Mileage, Engine and New Price

Untitled0.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Files

- data\_raw
  - train.csv
  - sample\_data
  - derivedcolumn.csv
  - encoded\_data.csv
  - preprocessing.csv

```

Price
dtype: int64
0

[8] # removing units
df['Mileage'] = df['Mileage'].str.extract('(\d+\.\d+)').astype(float)
df['Engine'] = df['Engine'].str.replace(' CC', '').astype(float)
df['Power'] = df['Power'].str.extract('(\d+\.\d+)').astype(float)
df['New_Price'] = df['New_Price'].str.extract('(\d+\.\d+)').astype(float)

df.head()

```

Unnamed: 0	Name	Location	Year	Kilometers_Driven	Fuel_Type	Transmission	Owner_Type	Mileage	Engine	Power	Seats	New_Price	Price	
0	1	Hyundai Creta 1.6 CRDi SX Option	Pune	2015	41000	Diesel	Manual	First	19.67	1582.0	126.20	5.0	NaN	12.50
1	2	Honda Jazz V	Chennai	2011	46000	Petrol	Manual	First	NaN	1199.0	88.70	5.0	8.61	4.50
2	3	Maruti Ertiga VDI	Chennai	2012	87000	Diesel	Manual	First	20.77	1248.0	88.76	7.0	NaN	6.00
3	4	Audi A4 New 2.0 TDI Multitronic	Coimbatore	2013	40670	Diesel	Automatic	Second	15.20	1968.0	140.80	5.0	NaN	17.74
4	6	Nissan Micra Diesel XV	Jaipur	2013	86999	Diesel	Manual	First	23.08	1461.0	63.10	5.0	NaN	3.50

Next steps: [Generate code with df](#) [View recommended plots](#)

4. Handling the Null values by removing the entire column “New\_Price” because it contains so many null values.

Untitled0.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Files

- data\_raw
  - train.csv
  - sample\_data
  - derivedcolumn.csv
  - encoded\_data.csv
  - preprocessing.csv

```

[ ] # Handling Null Values

df['Engine'].fillna(df['Engine'].median(), inplace=True)
df['Engine'].fillna(df['Power'].mean(), inplace=True)
df['Engine'].fillna(df['Mileage'].median(), inplace=True)
df['Engine'].fillna(df['Seats'].mode([0]), inplace=True)

[ ] # removing the entire column New_Price, because of so many null values

df.drop(columns=['New_Price'], inplace=True)

df.head()

```

Unnamed: 0	Name	Location	Year	Kilometers_Driven	Fuel_Type	Transmission	Owner_Type	Mileage	Engine	Power	Seats	Price	
0	1	Hyundai Creta 1.6 CRDi SX Option	Pune	2015	41000	Diesel	Manual	First	19.67	1582.0	126.20	5.0	12.50
1	2	Honda Jazz V	Chennai	2011	46000	Petrol	Manual	First	NaN	1199.0	88.70	5.0	4.50
2	3	Maruti Ertiga VDI	Chennai	2012	87000	Diesel	Manual	First	20.77	1248.0	88.76	7.0	6.00
3	4	Audi A4 New 2.0 TDI Multitronic	Coimbatore	2013	40670	Diesel	Automatic	Second	15.20	1968.0	140.80	5.0	17.74
4	6	Nissan Micra Diesel XV	Jaipur	2013	86999	Diesel	Manual	First	23.08	1461.0	63.10	5.0	3.50

Next steps: [Generate code with df](#) [View recommended plots](#)

```

[16] output_csv_file = 'preprocessing.csv' #preprocessing_data
df.to_csv(output_csv_file, index=False)

```

5. Encoding the columns Fuel Types, Transmission, Owner Type and Location to numerical by using 1 hot encoding.

Untitled0.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

Files

- data\_raw
  - train.csv
  - sample\_data
    - derivedcolumn.csv
    - encoded\_data.csv
    - preprocessing.csv

Code

```
encoded_df = pd.get_dummies(df, columns=['Fuel_Type', 'Transmission', 'Owner_Type', 'Location'])
```

df.head()

Unnamed: 0		Name	Location	Year	Kilometers_Driven	Fuel_Type	Transmission	Owner_Type	Mileage	Engine	Power	Seats	Price
0	1	Hyundai Creta 1.6 CRDi SX Option	Pune	2015	41000	Diesel	Manual	First	19.67	1582.0	126.20	5.0	12.50
1	2	Honda Jazz V	Chennai	2011	46000	Petrol	Manual	First	NaN	1199.0	88.70	5.0	4.50
2	3	Maruti Ertiga VDI	Chennai	2012	87000	Diesel	Manual	First	20.77	1248.0	88.76	7.0	6.00
3	4	Audi A4 New 2.0 TDI Multitronic	Coimbatore	2013	40670	Diesel	Automatic	Second	15.20	1968.0	140.80	5.0	17.74
4	6	Nissan Micra Diesel XV	Jaipur	2013	86999	Diesel	Manual	First	23.08	1461.0	63.10	5.0	3.50

Next steps: [Generate code with df](#) [View recommended plots](#)

```
[24] df2 = pd.get_dummies(df, columns=['Fuel_Type', 'Transmission', 'Owner_Type'], drop_first=True)
```

df2.head()

Unnamed: 0		Name	Location	Year	Kilometers_Driven	Mileage	Engine	Power	Seats	Price	Fuel_Type_Electric	Fuel_Type_Petrol	Transmission_Manual	Owner_Type
0	1	Hyundai Creta 1.6 CRDi SX Option	Pune	2015	41000	19.67	1582.0	126.20	5.0	12.50	0	0	1	First
1	2	Honda Jazz V	Chennai	2011	46000	NaN	1199.0	88.70	5.0	4.50	0	1	1	First
2	3	Maruti Ertiga VDI	Chennai	2012	87000	20.77	1248.0	88.76	7.0	6.00	0	0	1	First

Disk 81.40 GB available

## 6. And then Downloading the encoded csv file.

Untitled0.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

Files

- data\_raw
  - train.csv
  - sample\_data
    - derivedcolumn.csv
    - encoded\_data.csv
    - preprocessing.csv

Code

```
[24] df2 = pd.get_dummies(df, columns=['Fuel_Type', 'Transmission', 'Owner_Type'], drop_first=True)
```

df2.head()

Unnamed: 0		Name	Location	Year	Kilometers_Driven	Mileage	Engine	Power	Seats	Price	Fuel_Type_Electric	Fuel_Type_Petrol	Transmission_Manual	Owner_Type
0	1	Hyundai Creta 1.6 CRDi SX Option	Pune	2015	41000	19.67	1582.0	126.20	5.0	12.50	0	0	1	First
1	2	Honda Jazz V	Chennai	2011	46000	NaN	1199.0	88.70	5.0	4.50	0	1	1	First
2	3	Maruti Ertiga VDI	Chennai	2012	87000	20.77	1248.0	88.76	7.0	6.00	0	0	1	First
3	4	Audi A4 New 2.0 TDI Multitronic	Coimbatore	2013	40670	15.20	1968.0	140.80	5.0	17.74	0	0	0	Second
4	6	Nissan Micra Diesel XV	Jaipur	2013	86999	23.08	1461.0	63.10	5.0	3.50	0	0	1	First

Next steps: [Generate code with df2](#) [View recommended plots](#)

```
[27] output_csv_file = 'encoded_data.csv' #encoded_data
df.to_csv(output_csv_file, index=False)
```

Disk 81.40 GB available

## 7. Finding the current age of the vehicle to derive the new column "Mileage per year".

Untitled0.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Files

- data\_raw
- train.csv
- sample\_data
- derivedcolumn.csv
- encoded\_data.csv
- preprocessing.csv

[28] # Finding the current age of the vehicle

```
from datetime import datetime
current_year = datetime.now().year
df['Current_age'] = current_year - df['Year']
```

[29] df.head()

Unnamed: 0		Name	Location	Year	Kilometers_Driven	Fuel_Type	Transmission	Owner_Type	Mileage	Engine	Power	Seats	Price	Current_age
0	1	Hyundai Creta 1.6 CRDi SX Option	Pune	2015	41000	Diesel	Manual	First	19.67	1582.0	126.20	5.0	12.50	9
1	2	Honda Jazz V	Chennai	2011	46000	Petrol	Manual	First	NaN	1199.0	88.70	5.0	4.50	13
2	3	Maruti Ertiga VDI	Chennai	2012	87000	Diesel	Manual	First	20.77	1248.0	88.76	7.0	6.00	12
3	4	Audi A4 New 2.0 TDI Multitronic	Coimbatore	2013	40670	Diesel	Automatic	Second	15.20	1968.0	140.80	5.0	17.74	11
4	6	Nissan Micra Diesel XV	Jaipur	2013	86999	Diesel	Manual	First	23.08	1461.0	63.10	5.0	3.50	11

Next steps: [Generate code with df](#) [View recommended plots](#)

## 8. Deriving the mileage per year and then downloading the csv file.

Untitled0.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Files

- data\_raw
- train.csv
- sample\_data
- derivedcolumn.csv
- encoded\_data.csv
- preprocessing.csv

[30] # deriving the Mileage per year

```
df['Mileage_per_year'] = df['Mileage'] / df['Current_age']
```

df.head()

Unnamed: 0		Name	Location	Year	Kilometers_Driven	Fuel_Type	Transmission	Owner_Type	Mileage	Engine	Power	Seats	Price	Current_age	Mileage_per_y
0	1	Hyundai Creta 1.6 CRDi SX Option	Pune	2015	41000	Diesel	Manual	First	19.67	1582.0	126.20	5.0	12.50	9	2.18
1	2	Honda Jazz V	Chennai	2011	46000	Petrol	Manual	First	NaN	1199.0	88.70	5.0	4.50	13	
2	3	Maruti Ertiga VDI	Chennai	2012	87000	Diesel	Manual	First	20.77	1248.0	88.76	7.0	6.00	12	1.73
3	4	Audi A4 New 2.0 TDI Multitronic	Coimbatore	2013	40670	Diesel	Automatic	Second	15.20	1968.0	140.80	5.0	17.74	11	1.38
4	6	Nissan Micra Diesel XV	Jaipur	2013	86999	Diesel	Manual	First	23.08	1461.0	63.10	5.0	3.50	11	2.09

Next steps: [Generate code with df](#) [View recommended plots](#)

output\_csv\_file = 'derivedcolumn.csv' # Derived Column  
df.to\_csv(output\_csv\_file, index=False)

Disk 81.40 GB available

## 9. Manipulations of given data set like group by...

1Data Analysis.ipynb

File Edit View Insert Runtime Tools Help

All changes saved

Comment Share Colab AI

Files

data\_raw

train.csv

sample\_data

README.md

anscombe.json

california\_housing\_test.csv

california\_housing\_train.csv

mnist\_test.csv

mnist\_train\_small.csv

derivedcolumn.csv

encoded\_data.csv

preprocessing.csv

[21] output\_csv\_file = 'derivedcolumn.csv' # Derived Column

df.to\_csv(output\_csv\_file, index=False)

average\_price\_by\_year\_fuel = df.groupby(['Year', 'Fuel\_Type'])['Price'].mean().reset\_index()

# 2. Find the location with the highest average price

location\_max\_avg\_price = df.groupby('Location')['Price'].mean().idxmax()

# 3. Calculate the total kilometers driven by owner type

total\_kms\_by\_owner\_type = df.groupby('Owner\_Type')['Kilometers\_Driven'].sum()

# 5. Calculate the median mileage by transmission type and fuel type

median\_mileage\_by\_transmission\_fuel = df.groupby(['Transmission', 'Fuel\_Type'])['Mileage'].median().reset\_index()

# Printing the results

print("\nAverage Price by Year and Fuel Type:")

print(average\_price\_by\_year\_fuel.head())

print("\nLocation with the Highest Average Price:")

print(location\_max\_avg\_price)

print("\nTotal Kilometers Driven by Owner Type:")

print(total\_kms\_by\_owner\_type)

print("\nMedian Mileage by Transmission Type and Fuel Type:")

print(median\_mileage\_by\_transmission\_fuel.head())

Average Price by Year and Fuel Type:

Year Fuel\_Type Price

0 1998 Diesel 3.900

1 1998 Petrol 0.490

2 1999 Petrol 0.835

3 2000 Diesel 1.725

4 2000 Petrol 0.625

1Data Analysis.ipynb

File Edit View Insert Runtime Tools Help

All changes saved

Comment Share Colab AI

Files

data\_raw

train.csv

sample\_data

README.md

anscombe.json

california\_housing\_test.csv

california\_housing\_train.csv

mnist\_test.csv

mnist\_train\_small.csv

derivedcolumn.csv

encoded\_data.csv

preprocessing.csv

print(location\_max\_avg\_price)

print("\nTotal Kilometers Driven by Owner Type:")

print(total\_kms\_by\_owner\_type)

print("\nMedian Mileage by Transmission Type and Fuel Type:")

print(median\_mileage\_by\_transmission\_fuel.head())

Average Price by Year and Fuel Type:

Year Fuel\_Type Price

0 1998 Diesel 3.900

1 1998 Petrol 0.490

2 1999 Petrol 0.835

3 2000 Diesel 1.725

4 2000 Petrol 0.625

Location with the Highest Average Price:

Coimbatore

Total Kilometers Driven by Owner Type:

Owner\_Type

First 265534977

Fourth & Above 994833

Second 65837418

Third 9156829

Name: Kilometers\_Driven, dtype: int64

Median Mileage by Transmission Type and Fuel Type:

Transmission Fuel\_Type Mileage

0 Automatic Diesel 16.00

1 Automatic Electric NaN

2 Automatic Petrol 15.60

3 Manual Diesel 20.77

4 Manual Petrol 18.40