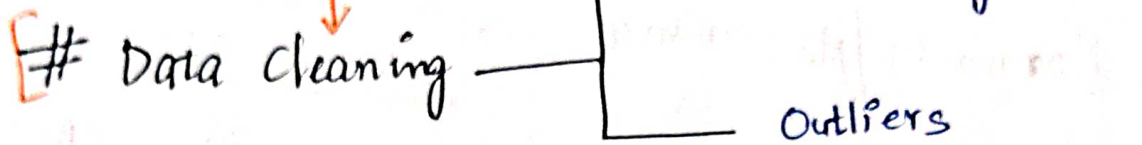


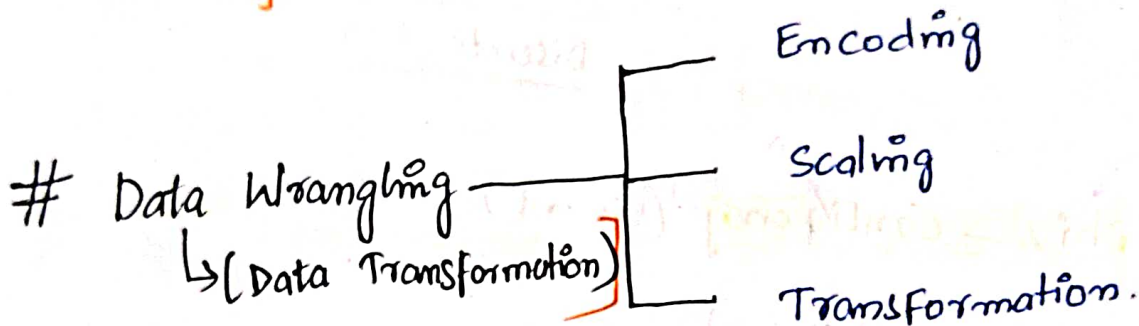
DL: 2/4/22
5:50pm

* Data Preprocessing / Data Preparation *

⇒ Feature Engineering



+



Us.
(Car Insurance) # Data cleaning (In spyder)

import numpy as np

import pandas as pd

why two Dataset
Because to check old
and new dataset.

⇒ Load Data Set

df_raw = pd.read_csv("claimants.csv")

df = pd.read_csv("claimants.csv")

[Out]: df_raw | dataframe | 1340, 7 | column name: CASENUM,
df | dataframe | 1340, 7 | column names: "

⇒ Check is there null values.

df.isnull().sum()

Out :-	CASENUM	0
	CLMSEX	12
	CLMINSUR	41
	SEATBELT	48
	CLMAGE	189
	LOSS	0
	ATTORNEY	0

MISSING Values.

Data Understanding

Case Num :- Number of case (insurance)

Discrete clm sex :- 0, 1 [Male] [Female] Person who is

Discrete clm insur :- 1 (Yes), 0 [No] # having insurance or not claiming

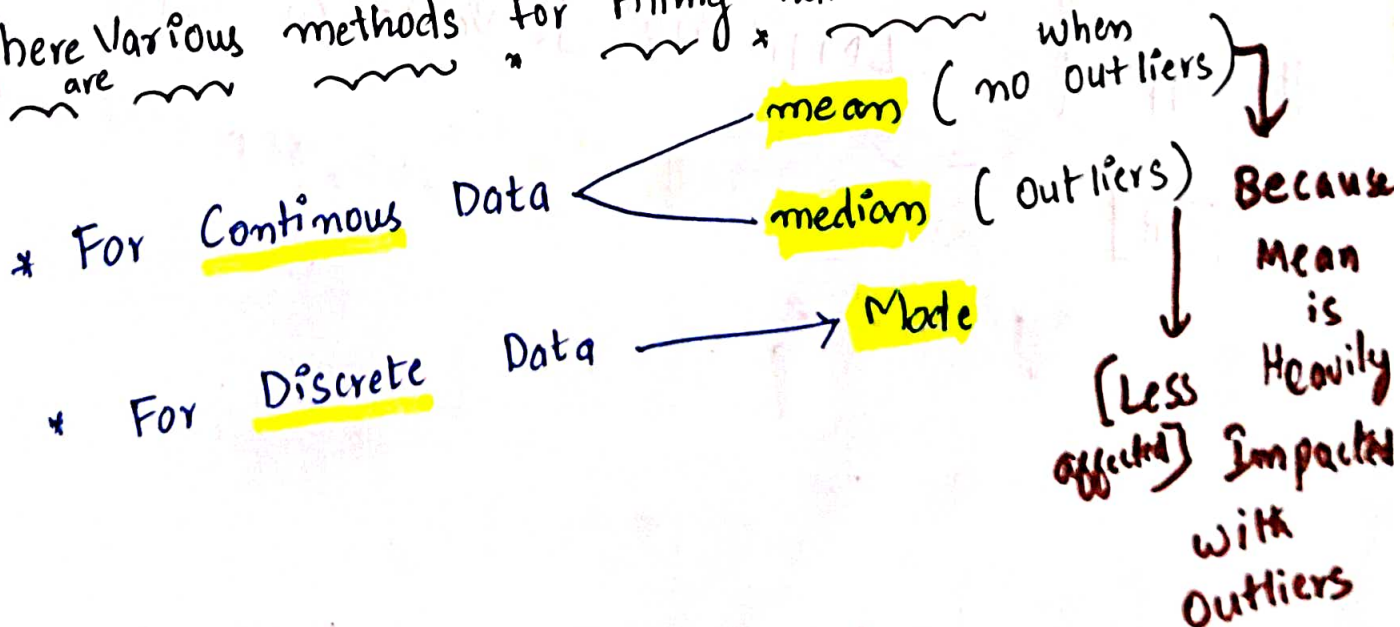
Discrete Seat belt :- 0 [No], 1 [Yes] # wearing seat belt Insurance is called "claimants" During The accident

Continuous clm age :- Age of The person

Loss :- Loss / Damage happened

attorney :- Whether They hire a lawyer (or) not, [0] No, [1] Yes.

There Various methods For Filling null values



Replacing null values For discrete Variables

So, we have to Replace Them with "mode"

df["CLMSEX"].value_counts()
(MODE)

Out :-
1.0 742
0.0 586
→ Highest, so we Replace with "1"

df["CLMSEX"].fillna(1, inplace=True)

↓
changing in original
Data = TRUE

Out Filled with "1"

df["CLMINSUR"].value_counts()

Out :-
1.0 1220
0.0 120
→ MODE

df["CLMINSUR"].fillna(1, inplace=True)

Out Filled with "1" - so, replaced with "1" value


```
# df["SEATBELT"].value_counts()
```

```
Out: 0.0 1270  
      1.0 22
```

→ Mode.

```
# df["SEATBELT"].fillna(0, inplace=True)
```

```
Out: Filled with "0"
```

Time
11:30pm

Replacing null values For Continuous Variables
By # Pandas

```
# df["CLMAGE"].mean()
```

```
Out: 28.414
```

```
# df["CLMAGE"].median()
```

```
Out: 30.00
```

```
# df["CLMAGE"].fillna(28.414, inplace=True)
```

```
Out: Filled with 28.414
```

If we ask again what is median value.

```
# df["CLMAGE"].median()
```

```
Out:
```

28.414

, so it changes 30.00 To 28.414
Because we Filled with null values
28.414.

2nd method

sklearn (skit Learn) (one stop shop for ml)

* SIMPLE IMPUTER

Load dataset again
Library → (module) (Function)

From sklearn.impute import SimpleImputer

Ex: missing values = "???"

mean_imputer = SimpleImputer (strategy = "mean")
(Function) → (Argument)

name as
Function. (or) store in
mean_imputer

EX:- Core Python

[Keyword argument]

If we don't write this line it gives

Same: $a = 4$

Converting array to data frame

To Remove array from mean value.

df["CLMAGE"] = Pd.DataFrame (mean_imputer.fit_trans

array([50.,
18.,
5.,
...])

storing

- Trans

Form (df[["CLMAGE"]])

Fill in this column (or) Replacing

Calculate The mean value

Out

Filled with 28.414

For Median only change mean To median

median_imputer = SimpleImputer (strategy = "Median")


```
# df["CLMAGE"] = pd.DataFrame (median_imputer.fit_
Transform (df[["CLMAGE"]]))
```

For Mode [Discrete Data]

```
mode_imputer = SimpleImputer (strategy="mode")
```

```
# df["CLMSEX"] = pd.DataFrame (mode_imputer.fit_transform(df[
"CLMSEX"])]
```

```
# df["CLMINSUR"] = pd.DataFrame (mode_imputer.fit_transfo
(df[["CLMINSUR"]]))
```

```
# df["SEATBELT"] = pd.DataFrame (mode_imputer.fit_trans
m(df[["SEATBELT"]]))
```

Output Fills Directly The mode

Value in to the dataset.

MODE = it is for Discrete Data.

MEAN
MEDIAN } → Continuous Data