Assignment - 2

1) Given, $s.d = 1.5$, $\bar{x} = 2.5$

The formula to calculate z-score,

$$z = \frac{x_i - \mu}{s.d.}$$

∴ For $x = 2$

$$z_1 = \frac{2 - 2.5}{1.5} = -0.33$$

For $x = 3$

$$z_2 = \frac{3 - 2.5}{1.5} = 0.33$$

For $x = 1$

$$z_3 = \frac{1 - 2.5}{1.5} = -1$$

For $x = 3$

$$z_4 = \frac{3 - 2.5}{1.5} = 0.33$$

For $x = 4$

$$z_5 = \frac{4 - 2.5}{1.5} = 1$$

Formula for Normalization -

$$\text{Normalization} = \frac{x - x_{min}}{x_{max} - x_{min}}$$

2) One-Hot encoding -

- One-hot-encoding is a data preprocessing step to convert categorical values into compatible numerical representation.
- Pandas function which performs OHE is get_dummies function.

3) - Function transformers
  - i) log transformer
  - ii) Reciprocal transformer
  - iii) Square or square root transformer
  - iv) Custom transformer
- Power transformers
  - i) box cox
  - ii) Yeo johnson

4) Assumptions of linear regression

i) Linearity - The relationship between each independent variable (x) and dependent variable (Y), is assumed to be linear.

ii) Homoscedasticity - The variance of the errors (ie. residuals) should be constant across all levels of the independent variables.

iii) Normality of Residuals : The residuals should be normally distributed

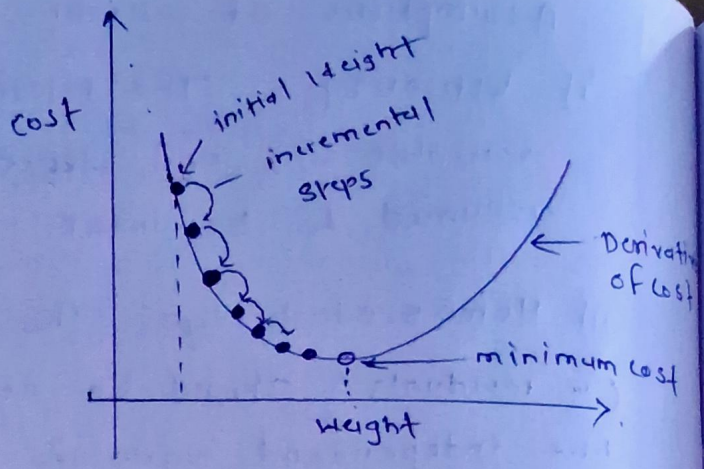
iv) Independence - The observations are assumed to be independent of each other, ie. the value of dependent variable for one obs^n should not be influenced by the value of dependent variable's obs^n.

v) Lack of Multicollinearity - In multiple linear regression there should not be perfect linear relationship among the independent variables.

5) Gradient Descent is used to minimize the cost function or loss function during training. The goal of gradient descent is to find the minimum of a fun^n by iteratively moving in the direction of the steepest decrease in the function.

To find the local minimum of a function using gradient descent, we must take steps proportional to the negative of the gradient

(move away from the gradient) of the function at the current point.



6) Pandas profiling is a powerful python library for data analysis and exploration. It provides a comprehensive report of dataset, allowing you to quickly understand the structure and properties of your data.

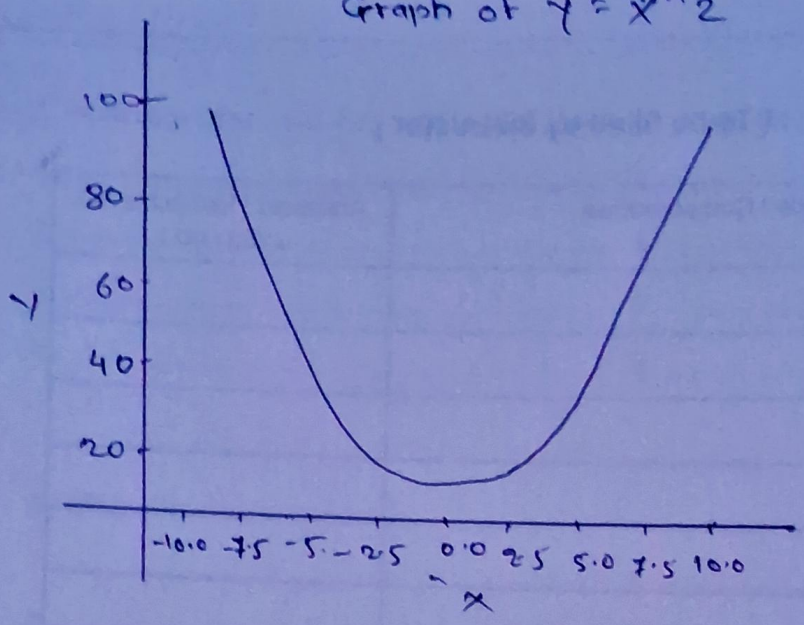
Syntax -

```
pip install ydata-profiling
from ydata_profiling import ProfileReport
Prof = ProfileReport( )
prof.to_file (output_file = "name of file.html")
```

7) Equation - $y = x^2$

We have to generate the values for x.

```
import numpy as np
x = np.linspace (-10,10,100)
Y = x**2
import matplotlib.pyplot as plt.
plt.plot (x,y, label = 'Y=x**2')
plt.xlabel ('x')
plt.ylabel ('y')
plt.title ('Graph of y=x^2')
plt.legend ()
plt.show ()
```



8)

- import necessary libraries

```
import seaborn as sns
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error
import matplotlib.pyplot as plt.
```

- import dataset from seaborn library

```
mpg = sns.load_dataset('mpg')
```

- check for missing value.

```
mpg.isnull().sum()
```

```
features = mpg.select_dtypes(include=['float64',
                                      'int64']).dropna()
X = features.drop('mpg', axis=1)
Y = features['mpg']

X_train, x_test, X_train, Y_test = train_test_split(x, Y,
                                   test_size=0.2, random_state=42)
model = LinearRegression()
model.fit(X_train, Y_train)
Y_pred = model.predict(X_test)

MSE = mean_squared_error(Y_test, Y_pred)
PRINT(MSE)
→ 10.50
```