

```
import pandas as pd
```

```
import numpy as np
```

```
df = pd.read_csv(r'https://github.com/YBI-Foundation/Dataset/raw/main/Admission%20Ch
```

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 400 entries, 0 to 399
Data columns (total 9 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Serial No             400 non-null   int64
1   GRE Score             400 non-null   int64
2   TOEFL Score           400 non-null   int64
3   University Rating     400 non-null   int64
4   SOP                   400 non-null   float64
5   LOR                   400 non-null   float64
6   CGPA                  400 non-null   float64
7   Research              400 non-null   int64
8   Chance of Admit       400 non-null   float64
dtypes: float64(4), int64(5)
memory usage: 28.2 KB
```

```
df.head()
```

| | Serial No | GRE Score | TOEFL Score | University Rating | SOP | LOR | CGPA | Research | Chance of Admit |
|----------|--------------|--------------|----------------|----------------------|-----|-----|------|----------|--------------------|
| 0 | 1 | 337 | 118 | 4 | 4.5 | 4.5 | 9.65 | 1 | 0.92 |
| 1 | 2 | 324 | 107 | 4 | 4.0 | 4.5 | 8.87 | 1 | 0.76 |
| 2 | 3 | 316 | 104 | 3 | 3.0 | 3.5 | 8.00 | 1 | 0.72 |
| 3 | 4 | 322 | 110 | 3 | 3.5 | 2.5 | 8.67 | 1 | 0.80 |
| 4 | 5 | 314 | 103 | 2 | 2.0 | 3.0 | 8.21 | 0 | 0.65 |

```
df.describe()
```

| | Serial No | GRE Score | TOEFL Score | University Rating | SOP | LOR | CG |
|--------------|------------|------------|-------------|-------------------|------------|------------|----------|
| count | 400.000000 | 400.000000 | 400.000000 | 400.000000 | 400.000000 | 400.000000 | 400.0000 |

```

y = df['Chance of Admit ']
std    115.614301    11.473646    6.069514    1.143728    1.006869    0.898478    0.5963
df.columns

Index(['Serial No', 'GRE Score', 'TOEFL Score', 'University Rating', ' SOP',
      'LOR ', 'CGPA', 'Research', 'Chance of Admit '],
      dtype='object')

75%    300 2500000    325 0000000    112 0000000    4 0000000    4 0000000    4 0000000    9 0625
X = df[['GRE Score', 'TOEFL Score', 'University Rating', ' SOP',
      'LOR ', 'CGPA', 'Research']]

from sklearn.model_selection import train_test_split

X_train,X_test,y_train,y_test = train_test_split(X,y,test_size=0.3,random_state = 25

X_train.shape,X_test.shape,y_train.shape,y_test.shape

((280, 7), (120, 7), (280,), (120,))

from sklearn.preprocessing import StandardScaler

sc = StandardScaler()

X_train= sc.fit_transform(X_train)

X_test = sc.fit_transform(X_test)

X_train= pd.DataFrame(X_train)

X_test = pd.DataFrame(X_test)

X_train

```

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 |
|-----|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
| 0 | 0.147424 | -0.462792 | -0.118512 | -0.429673 | 0.047705 | 0.086351 | 0.866025 |
| 1 | 0.319060 | 0.359427 | -0.118512 | 0.073710 | 1.717371 | 0.383422 | 0.866025 |
| 2 | -0.453305 | -0.133904 | -0.118512 | -0.429673 | -1.621962 | -1.184452 | 0.866025 |
| 3 | -0.624942 | -0.133904 | -0.118512 | 0.073710 | 0.047705 | 0.086351 | -1.154701 |
| 4 | 0.319060 | 0.194983 | -0.118512 | -0.429673 | -0.508851 | -0.128200 | 0.866025 |
| ... | ... | ... | ... | ... | ... | ... | ... |
| 275 | 0.319060 | 0.523871 | 1.675190 | 1.583859 | 1.717371 | 1.373658 | 0.866025 |

X_test

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 |
|-----|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
| 0 | -0.155745 | 0.090237 | -0.821428 | -0.812236 | 0.623938 | 0.008505 | -0.983470 |
| 1 | 0.393943 | -0.417347 | 0.013923 | 0.157598 | 1.182688 | -0.378613 | 1.016808 |
| 2 | 0.119099 | -0.248152 | 0.013923 | 0.157598 | -0.493563 | 0.008505 | -0.983470 |
| 3 | -0.430589 | -1.263321 | -1.656778 | -0.812236 | -0.493563 | -0.220246 | 1.016808 |
| 4 | -1.529965 | -1.094126 | -0.821428 | -0.327319 | 0.065188 | -1.188040 | -0.983470 |
| ... | ... | ... | ... | ... | ... | ... | ... |
| 115 | 0.577173 | 0.428627 | 1.684623 | 1.127431 | 0.065188 | 0.430815 | -0.983470 |
| 116 | 0.210714 | -1.094126 | -0.821428 | -0.812236 | 0.065188 | -0.026688 | 1.016808 |
| 117 | -0.980277 | -0.078958 | -0.821428 | -0.327319 | -0.493563 | -0.695345 | -0.983470 |
| 118 | 1.126861 | 0.597821 | 0.849273 | 1.612348 | 0.623938 | 1.029087 | 1.016808 |
| 119 | -0.064130 | -0.078958 | 0.013923 | 1.127431 | 0.065188 | -0.237843 | -0.983470 |

120 rows × 7 columns

X_test

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 |
|---|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
| 0 | -0.155745 | 0.090237 | -0.821428 | -0.812236 | 0.623938 | 0.008505 | -0.983470 |
| 1 | 0.393943 | -0.417347 | 0.013923 | 0.157598 | 1.182688 | -0.378613 | 1.016808 |
| 2 | 0.119099 | -0.248152 | 0.013923 | 0.157598 | -0.493563 | 0.008505 | -0.983470 |
| 3 | -0.430589 | -1.263321 | -1.656778 | -0.812236 | -0.493563 | -0.220246 | 1.016808 |

```
from sklearn.linear_model import LinearRegression
```

```
...
```

```
lr = LinearRegression()
```

```
116 0.210714 -1.094126 -0.821428 -0.812236 0.065188 -0.026688 1.016808
```

```
lr.fit(X_train,y_train)
```

```
LinearRegression()
```

```
116 0.210714 -1.094126 -0.821428 -0.812236 0.065188 -0.026688 1.016808
```

```
lr.intercept_
```

```
0.733785714285714
```

```
lr.coef_
```

```
array([ 0.02377773,  0.01746937,  0.00632086, -0.00378001,  0.01772667,
        0.06855606,  0.01020204])
```

```
df.columns
```

```
Index(['Serial No', 'GRE Score', 'TOEFL Score', 'University Rating', 'SOP',
      'LOR ', 'CGPA', 'Research', 'Chance of Admit '],
      dtype='object')
```

► Regression_Equation

Chance Of admission = 0.7337857142857143+0.02377773(GRE score)+ 0.01746937(TOEFL score) +0.00632086(University Rating) -0.00378001(SOP)+ 0.01772667(LOR)+ 0.06855606(CGPA) + 0.01020204(Research)*

[] ↳ 3 cells hidden

► Stats Model

[] ↳ 2 cells hidden

▸ Pycaret

▸ R Language

[] ↵ 6 cells hidden

