```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
from sklearn import linear_model
import statsmodels
import statsmodels.api as sm
```

```
df=pd.read_excel('/content/EXCEL FILE 1.xlsx')
```

```
df
```

	Person ID	Gender	Age	Occupation	Sleep Duration	Quality of Sleep	Physical Activity Level	Stress Level	Cate
0	1	Male	27	Software Engineer	6.1	6	42	6	Over
1	2	Male	28	Doctor	6.2	6	60	8	N
2	3	Male	28	Doctor	6.2	6	60	8	N
3	4	Male	28	Sales Representative	5.9	4	30	8	
4	5	Male	28	Sales Representative	5.9	4	30	8	
...	...	...	...	...	...	...	...	...	
369	370	Female	59	Nurse	8.1	9	75	3	Over
370	371	Female	59	Nurse	8.0	9	75	3	Over
371	372	Female	59	Nurse	8.1	9	75	3	Over
372	373	Female	59	Nurse	8.1	9	75	3	Over

```
df.columns
```

```
Index(['Person ID', 'Gender', 'Age', 'Occupation', 'Sleep Duration',
       'Quality of Sleep', 'Physical Activity Level', 'Stress Level',
       'BMI Category', 'Blood Pressure', 'Heart Rate', 'Daily Steps',
       'Sleep Disorder'],
      dtype='object')
```

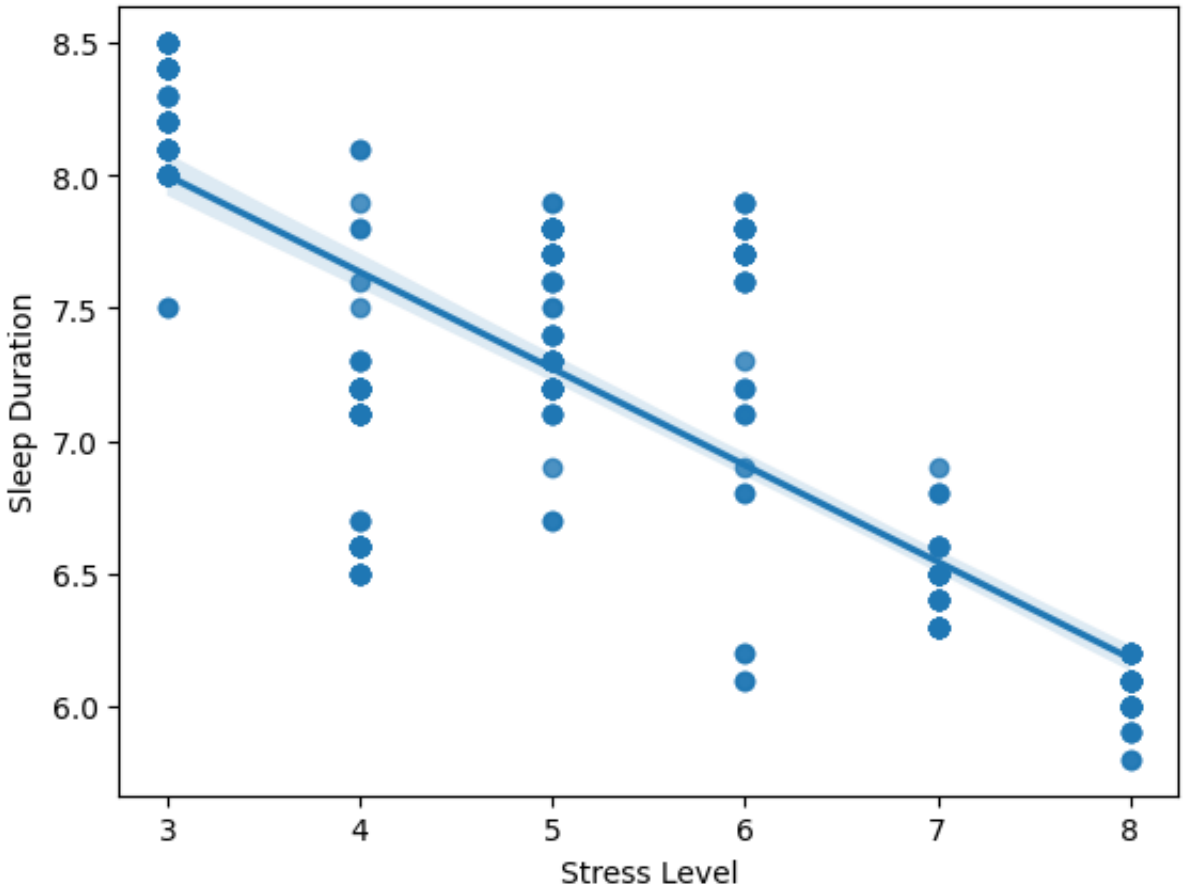
```
x=df[[ 'Sleep Duration' ]]
y=df['Stress Level']
```

```
df.describe()
```

	Person ID	Age	Sleep Duration	Quality of Sleep	Physical Activity Level	Stress Level	Hear Rat
count	374.000000	374.000000	374.000000	374.000000	374.000000	374.000000	374.00000
mean	187.500000	42.184492	7.132086	7.312834	59.171123	5.385027	70.16577
std	108.108742	8.673133	0.795657	1.196956	20.830804	1.774526	4.13567
min	1.000000	27.000000	5.800000	4.000000	30.000000	3.000000	65.00000
25%	94.250000	35.250000	6.400000	6.000000	45.000000	4.000000	68.00000
50%	187.500000	43.000000	7.200000	7.000000	60.000000	5.000000	70.00000
75%	280.750000	50.000000	7.800000	8.000000	75.000000	7.000000	72.00000

```
sns.regplot(x='Stress Level', y= 'Sleep Duration',data=df)
```

```
<Axes: xlabel='Stress Level', ylabel='Sleep Duration'>
```



```
df.corr()
```

```
<ipython-input-11-2f6f6606aa2c>:1: FutureWarning: The default value of numeric
  df.corr()
```

	Person ID	Age	Sleep Duration	Quality of Sleep	Physical Activity Level	Stress Level	Heart Rate	D S
Person ID	1.000000	0.990516	0.296305	0.431612	0.149882	-0.394287	-0.225467	0.0
Age	0.990516	1.000000	0.344709	0.473734	0.178993	-0.422344	-0.225606	0.0
Sleep Duration	0.296305	0.344709	1.000000	0.883213	0.212360	-0.811023	-0.516455	-0.0
Quality of Sleep	0.431612	0.473734	0.883213	1.000000	0.192896	-0.898752	-0.659865	0.0
Physical Activity Level	0.149882	0.178993	0.212360	0.192896	1.000000	-0.034134	0.136971	0.7

```
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test=train_test_split(x,y, test_size=0.2)
```

```
x_train_sm=sm.add_constant(x_train)
x_train_sm.head()
```

	const	Sleep Duration
316	1.0	8.5
328	1.0	8.3
294	1.0	6.1
104	1.0	7.2
370	1.0	8.0

```
lr=sm.OLS(y_train, x_train_sm)
lr_model=lr.fit()
lr_model.params
```

```
const             18.186836
Sleep Duration    -1.797411
dtype: float64
```

Linear regression equation = Stress Level = (-1.82) Sleep Duration + 18.40

If, I take 5 hours of sleep then 9.2

if I take 8 hours of sleep then 3.83

if I take 10 hours of sleep. then 0.25

```
lr_model.summary()
```

OLS Regression Results

Dep. Variable:	Stress Level	**R-squared:**	0.661
Model:	OLS	**Adj. R-squared:**	0.660
Method:	Least Squares	**F-statistic:**	579.4
Date:	Tue, 03 Oct 2023	**Prob (F-statistic):**	9.22e-72
Time:	19:18:25	**Log-Likelihood:**	-432.86
No. Observations:	299	**AIC:**	869.7
Df Residuals:	297	**BIC:**	877.1
Df Model:	1		
Covariance Type:	nonrobust		

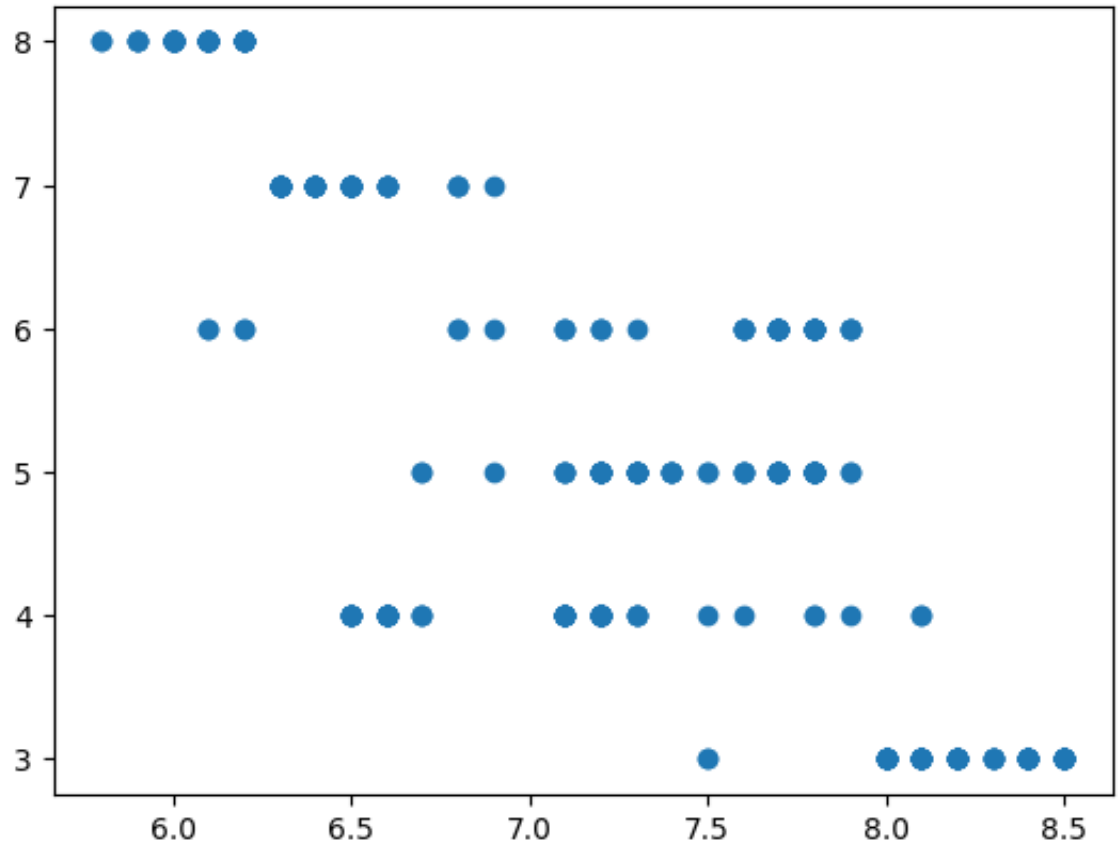
	coef	**std err**	**t**	**P>|t|**	**[0.025**	**0.975]**
const	18.1868	0.537	33.851	0.000	17.130	19.244
Sleep Duration	-1.7974	0.075	-24.071	0.000	-1.944	-1.650

Omnibus:	13.316	**Durbin-Watson:**	1.858
Prob(Omnibus):	0.001	**Jarque-Bera (JB):**	14.378
Skew:	-0.534	**Prob(JB):**	0.000755
Kurtosis:	2.889	**Cond. No.**	66.0

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

```
plt.scatter(x, y)
```

```
<matplotlib.collections.PathCollection at 0x7c9e58788130>
```



```
x_train
```

	Sleep Duration
316	8.5
328	8.3
294	6.1
104	7.2
370	8.0
...	...
51	7.5
135	7.3
94	7.2
254	6.5
0	6.1

299 rows × 1 columns

x_test

	Sleep Duration
237	6.5
369	8.1
8	7.8
5	5.9
43	7.8
...	...
285	6.0
270	6.1
364	8.0
236	6.4
3	5.9

75 rows × 1 columns

y_train

```
316    3
328    3
294    8
104    4
370    3
      ..
51     3
135    5
94     4
254    4
0      6
Name: Stress Level, Length: 299, dtype: int64
```

```
y_test
```

```
237    4
369    3
8      6
5      8
43     6
      ..
285    8
270    8
364    3
236    7
3      8
Name: Stress Level, Length: 75, dtype: int64
```

```
from sklearn.linear_model import LinearRegression
clf=LinearRegression()
clf.fit( x_train, y_train)
```

▾ LinearRegression

LinearRegression()

```
x_test
```

	Sleep Duration
237	6.5
369	8.1
8	7.8
5	5.9
43	7.8
...	...
285	6.0
270	6.1
364	8.0
236	6.4
3	5.9

75 rows × 1 columns

```
clf.predict(x_test)
```

```
array([6.50366683, 3.62780971, 4.16703292, 7.58211324, 4.16703292,
       5.42522041, 5.24547934, 5.24547934, 5.42522041, 3.44806864,
       5.06573827, 4.34677399, 6.32392576, 5.42522041, 5.24547934,
       6.50366683, 4.70625613, 3.62780971, 3.62780971, 4.34677399,
       7.40237217, 5.78470255, 7.2226311 , 3.44806864, 4.34677399,
       4.16703292, 7.2226311 , 6.50366683, 3.26832757, 5.24547934,
       6.50366683, 7.40237217, 7.2226311 , 6.50366683, 4.8859972 ,
       7.58211324, 7.2226311 , 5.96444362, 6.6834079 , 6.32392576,
       3.44806864, 6.50366683, 6.50366683, 6.32392576, 6.50366683,
       4.34677399, 6.32392576, 3.0885865 , 4.8859972 , 6.6834079 ,
       5.06573827, 4.34677399, 5.42522041, 5.24547934, 5.06573827,
       4.16703292, 5.24547934, 3.80755078, 7.2226311 , 7.04289003,
       3.0885865 , 6.50366683, 6.32392576, 7.2226311 , 2.90884543,
       3.62780971, 5.24547934, 4.70625613, 7.76185431, 4.16703292,
       7.40237217, 7.2226311 , 3.80755078, 6.6834079 , 7.58211324])
```

```
y_test
```

```
237    4
369    3
8      6
5      8
43     6
      ..
285    8
270    8
364    3
236    7
3      8
Name: Stress Level, Length: 75, dtype: int64
```

```
clf.score(x_test, y_test)
```

```
0.638286384175617
```

```
new_sleep_duration = [[6]]
```

```
predicted_stress_level = clf.predict(new_sleep_duration)
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/base.py:439: UserWarning: X do
  warnings.warn(
```

```
print("Predicted Stress Level", predicted_stress_level[0])
```

```
Predicted Stress Level 7.402372174377762
```