```python
In [1]: import pandas as pd
        from matplotlib import pyplot as plt
        %matplotlib inline
        import seaborn as sns
        import numpy as np
        from scipy import stats
        import statsmodels.formula.api as smf

        import warnings
        warnings.filterwarnings('ignore')
```

1)

```python
In [4]: delivery_data =pd.read_csv('delivery_time.csv')
        delivery_data
```

Out[4]:

	Delivery Time	Sorting Time
0	21.00	10
1	13.50	4
2	19.75	6
3	24.00	9
4	29.00	10
5	15.35	6
6	19.00	7
7	9.50	3
8	17.90	10
9	18.75	9
10	19.83	8
11	10.75	4
12	16.68	7
13	11.50	3
14	12.03	3
15	14.88	4
16	13.75	6
17	18.11	7
18	8.00	2
19	17.83	7
20	21.50	5

In [5]: `delivery_data.shape`

Out[5]: `(21, 2)`

In [6]: `delivery_data.isna().sum()`

Out[6]:
```
Delivery Time      0
Sorting Time       0
dtype: int64
```

In [7]: `delivery_data.dtypes`

Out[7]:
```
Delivery Time    float64
Sorting Time       int64
dtype: object
```

In [8]: `delivery_data.info(show_counts = all)`

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 21 entries, 0 to 20
Data columns (total 2 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   Delivery Time  21 non-null     float64
 1   Sorting Time   21 non-null     int64
dtypes: float64(1), int64(1)
memory usage: 464.0 bytes
```

In [9]:
```python
sns.distplot(a=delivery_data['Delivery Time'],hist=False)
plt.show()
```
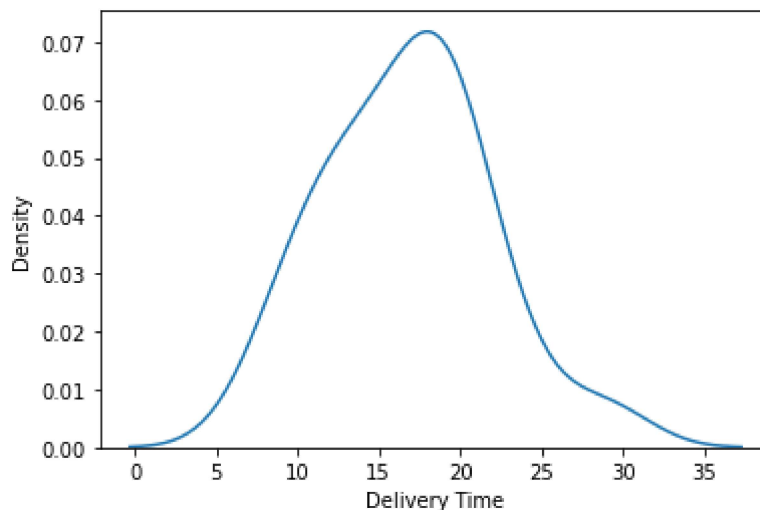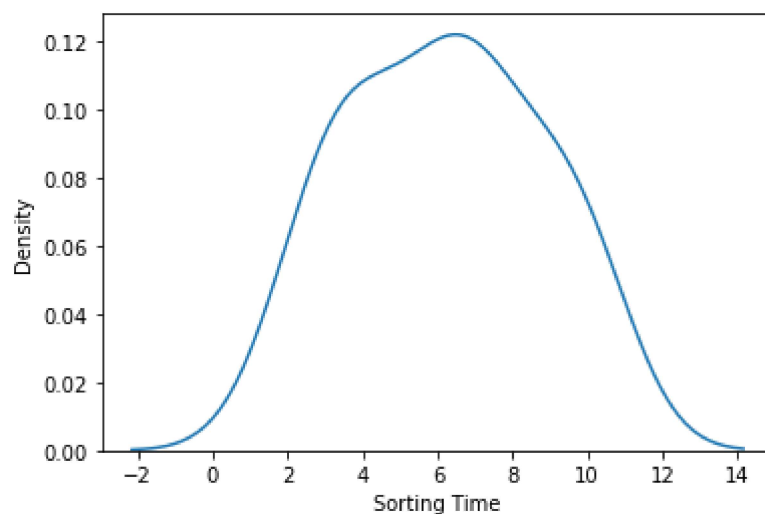

In [10]: `delivery_data['Delivery Time'].skew()`

Out[10]: `0.3523900822831107`

In [11]: `delivery_data['Delivery Time'].kurtosis()`

Out[11]: 0.31795982942685397

In [12]:
```python
sns.distplot(a=delivery_data['Sorting Time'],hist=False)
plt.show()
```


In [13]: `delivery_data['Sorting Time'].skew()`

Out[13]: 0.047115474210530174

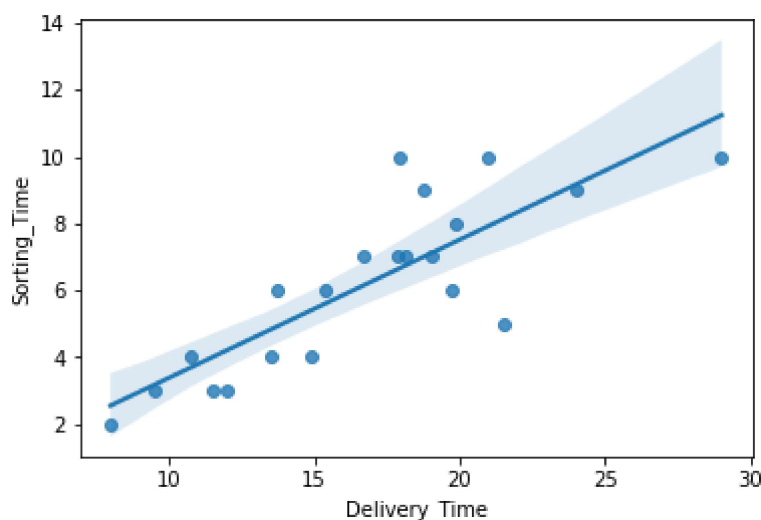In [14]: `delivery_data['Sorting Time'].kurtosis()`

Out[14]: -1.14845514534878

In [15]: `delivery_data = delivery_data.rename({'Delivery Time':'Delivery_Time','Sorting Ti`

In [16]: `delivery_data.corr()`

Out[16]:

	Delivery_Time	Sorting_Time
Delivery_Time	1.000000	0.825997
Sorting_Time	0.825997	1.000000

In [17]:
```python
sns.regplot(x=delivery_data['Delivery_Time'],y=delivery_data['Sorting_Time'])
plt.show()
```


In [18]:
```python
model = smf.ols(formula = 'Delivery_Time ~ Sorting_Time',data= delivery_data).fit
```

In [19]:
```python
model.params
```

Out[19]:
```
Intercept        6.582734
Sorting_Time     1.649020
dtype: float64
```

In [20]:
```python
model.pvalues,model.tvalues
```

Out[20]:
```
(Intercept        0.001147
 Sorting_Time     0.000004
 dtype: float64,
 Intercept        3.823349
 Sorting_Time     6.387447
 dtype: float64)
```

In [23]:
```python
round(model.rsquared,4),round(model.rsquared_adj,4)
```

Out[23]: (0.6823, 0.6655)

```
In [24]: delivery_time = 6.582734+(1.649020*6)
         delivery_time
```

Out[24]: 16.476854

```
In [29]: test_data = pd.DataFrame(data={'Sorting_Time':[5,6,7,8]})
```

```
In [30]: test_data
```

Out[30]:

	Sorting_Time
0	5
1	6
2	7
3	8

```
In [35]: model.predict(test_data)
```

Out[35]:
```
0    14.827833
1    16.476853
2    18.125873
3    19.774893
dtype: float64
```

2)

```
In [50]: data = pd.read_csv('Salary_Data.csv')
```

In [51]:   data

Out[51]:

	YearsExperience	Salary
0	1.1	39343.0
1	1.3	46205.0
2	1.5	37731.0
3	2.0	43525.0
4	2.2	39891.0
5	2.9	56642.0
6	3.0	60150.0
7	3.2	54445.0
8	3.2	64445.0
9	3.7	57189.0
10	3.9	63218.0
11	4.0	55794.0
12	4.0	56957.0
13	4.1	57081.0
14	4.5	61111.0
15	4.9	67938.0
16	5.1	66029.0
17	5.3	83088.0
18	5.9	81363.0
19	6.0	93940.0
20	6.8	91738.0
21	7.1	98273.0
22	7.9	101302.0
23	8.2	113812.0
24	8.7	109431.0
25	9.0	105582.0
26	9.5	116969.0
27	9.6	112635.0
28	10.3	122391.0
29	10.5	121872.0

```
In [52]: data.shape
```

```
Out[52]: (30, 2)
```

```
In [53]: data.isna().sum()
```

```
Out[53]: YearsExperience      0
         Salary               0
         dtype: int64
```

```
In [54]: data.info(show_counts ='all')
```
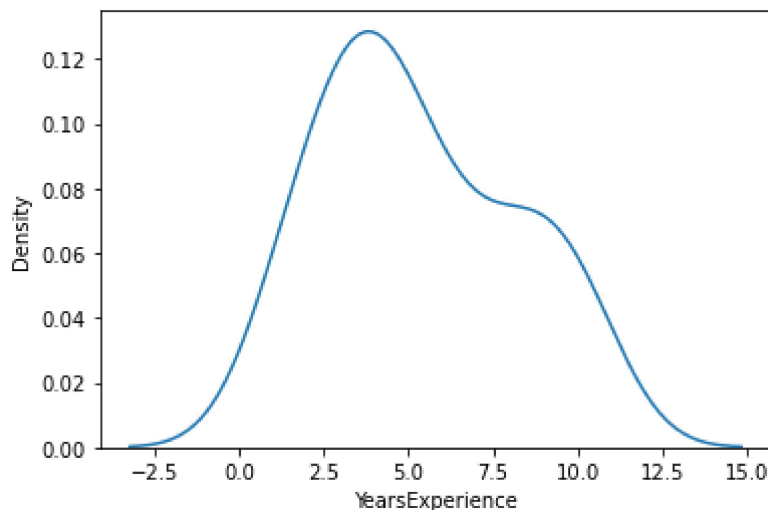
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 30 entries, 0 to 29
Data columns (total 2 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   YearsExperience  30 non-null    float64
 1   Salary           30 non-null    float64
dtypes: float64(2)
memory usage: 608.0 bytes
```

```
In [55]: data.dtypes
```

```
Out[55]: YearsExperience    float64
         Salary             float64
         dtype: object
```

```
In [56]: sns.distplot(a=data['YearsExperience'],hist = False)
         plt.show()
```
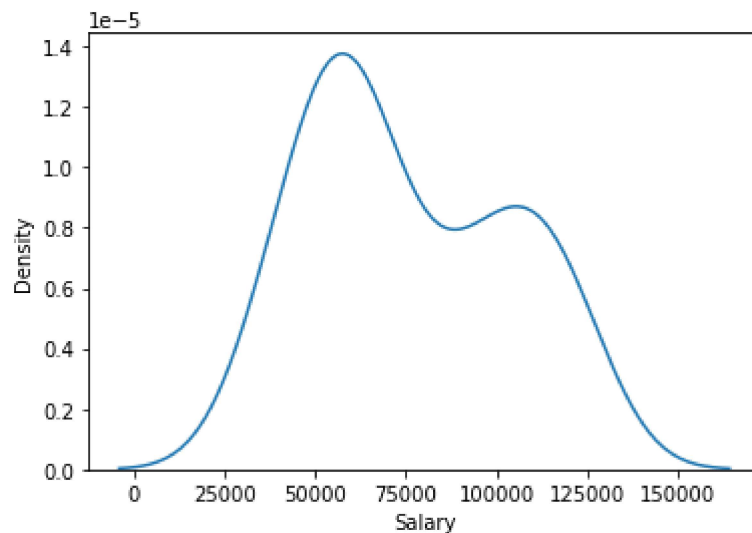


```
In [57]: data['YearsExperience'].skew()
```

```
Out[57]: 0.37956024064804106
```

In [58]: `data['YearsExperience'].kurt()`

Out[58]: -1.0122119403325072

In [59]:
```python
sns.distplot(a=data['Salary'],hist=False)
plt.show()
```


In [60]: `data['Salary'].skew()`

Out[60]: 0.35411967922959153

In [61]: `data['Salary'].kurt()`

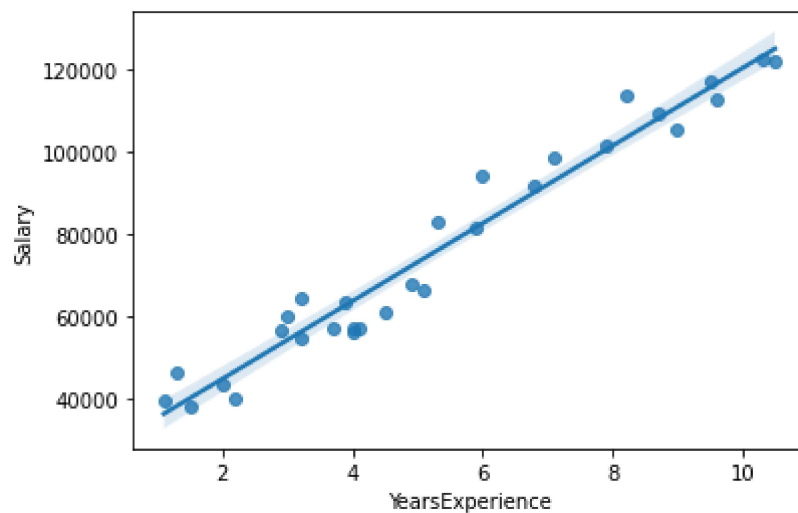Out[61]: -1.295421086394517

In [62]: `data.corr()`

Out[62]:

	YearsExperience	Salary
YearsExperience	1.000000	0.978242
Salary	0.978242	1.000000

In [63]:
```python
sns.regplot( x= data['YearsExperience'],y= data['Salary'])
plt.show()
```


In [77]:
```python
linear_model = smf.ols(formula = 'Salary~YearsExperience',data = data).fit()
```

In [65]:
```python
linear_model.params
```

Out[65]:
```
Intercept          25792.200199
YearsExperience     9449.962321
dtype: float64
```

In [66]:
```python
linear_model.tvalues
```

Out[66]:
```
Intercept          11.346940
YearsExperience    24.950094
dtype: float64
```

In [67]: `linear_model.pvalues`

Out[67]:
```
Intercept          5.511950e-12
YearsExperience    1.143068e-20
dtype: float64
```

In [68]: `round(linear_model.rsquared,4)`

Out[68]: `0.957`

In [69]: `round(linear_model.rsquared_adj,4)`

Out[69]: `0.9554`

In [70]:
```
salary_hike = 25792.2001+(9449.9623*3)
salary_hike
```

Out[70]: `54142.087`

In [78]:
```
salary_data_predct = pd.DataFrame(data = {'YearsExperience':[3,4,5,6,7]})
salary_data_predct
```

Out[78]:

	YearsExperience
0	3
1	4
2	5
3	6
4	7

In [79]: `linear_model.predict(salary_data_predct)`

Out[79]:
```
0    54142.087163
1    63592.049484
2    73042.011806
3    82491.974127
4    91941.936449
dtype: float64
```

In [ ]:

In [ ]: