

```
In [1]: import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
```

```
In [2]: marketing = pd.read_csv('/content/tvmarketing.csv')
```

```
In [3]: marketing.head()
```

Out[3]:

	TV	Sales
0	230.1	22.1
1	44.5	10.4
2	17.2	9.3
3	151.5	18.5
4	180.8	12.9

```
In [4]: marketing.tail()
```

Out[4]:

	TV	Sales
195	38.2	7.6
196	94.2	9.7
197	177.0	12.8
198	283.6	25.5
199	232.1	13.4

```
In [5]: marketing.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 200 entries, 0 to 199
Data columns (total 2 columns):
#   Column  Non-Null Count  Dtype
---  ---
0    TV      200 non-null    float64
1   Sales   200 non-null    float64
dtypes: float64(2)
memory usage: 3.2 KB
```

```
In [6]: marketing.shape
```

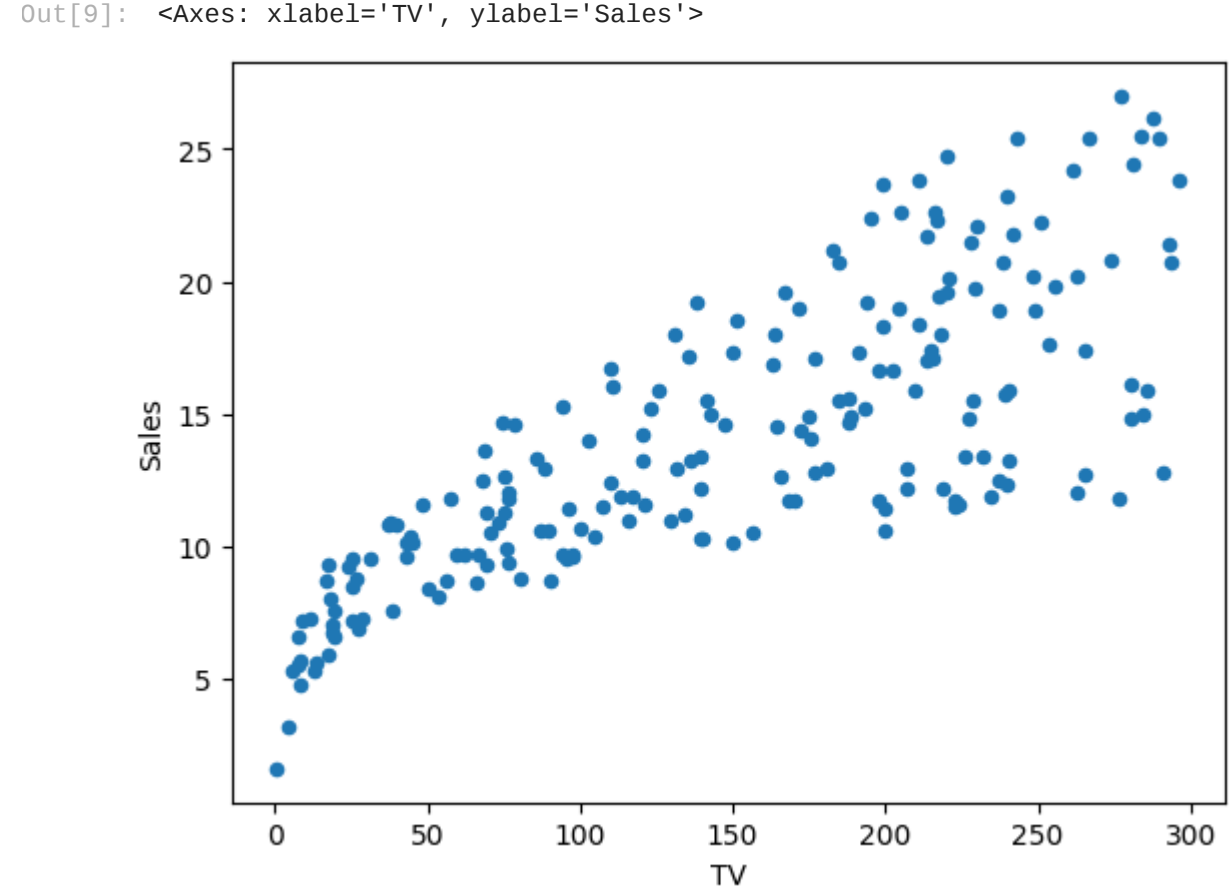
Out[6]: (200, 2)

```
In [8]: marketing.describe()
```

Out[8]:

	TV	Sales
count	200.000000	200.000000
mean	147.042500	14.022500
std	85.854236	5.217457
min	0.700000	1.600000
25%	74.375000	10.375000
50%	149.750000	12.900000
75%	218.825000	17.400000
max	296.400000	27.000000

```
In [9]: marketing.plot(x='TV',y='Sales',kind='scatter')
```



```
In [10]: from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn import metrics
```

```
In [11]: x = marketing["TV"].values.reshape(-1,1)
y = marketing['Sales'].values.reshape(-1,1)
```

```
In [12]: x_train,x_test,y_train,y_test = train_test_split(x,y,test_size=0.2,random_state =0)
```

```
In [13]: regressor = LinearRegression()
regressor.fit(x_train,y_train)
```

```
Out[13]: LinearRegression()
```

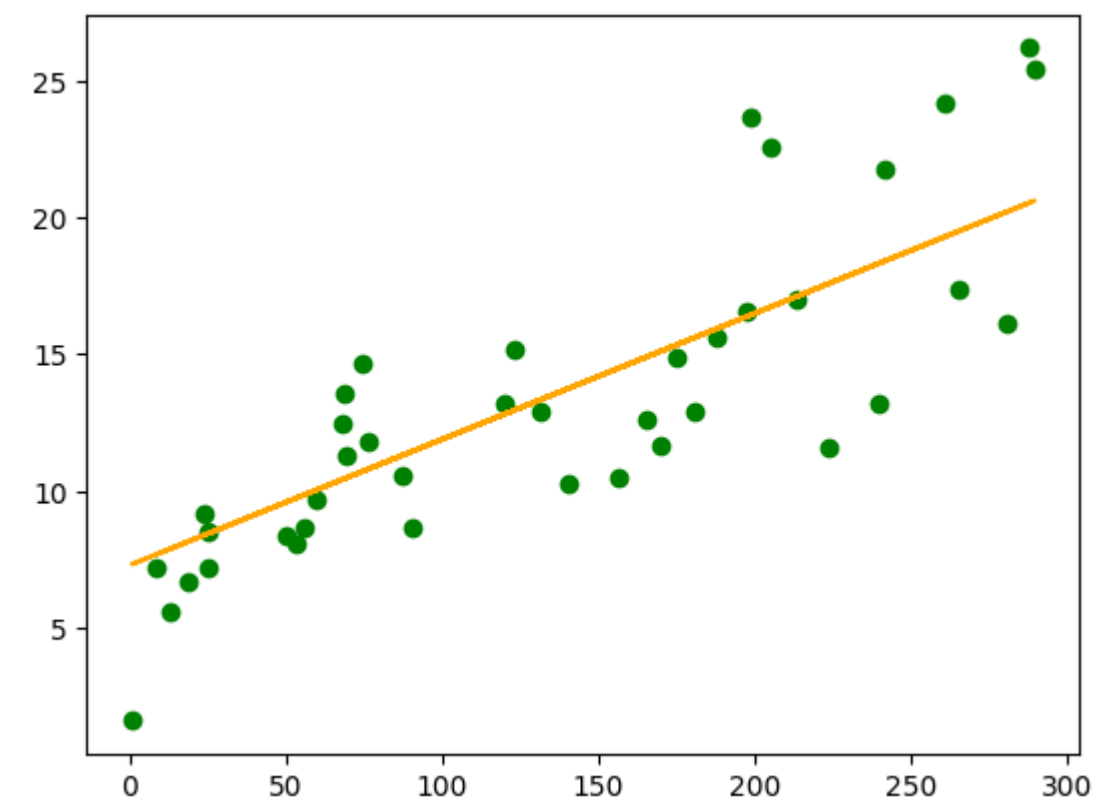
```
In [14]: y_pred = regressor.predict(x_test)
```

```
In [15]: print(regressor.intercept_)

print(regressor.coef_)

[7.29249377]
[[0.04600779]]
```

```
In [16]: plt.scatter(x_test,y_test,color ="green")
plt.plot(x_test,y_pred,color="orange")
plt.show()
```



```
In [17]: print("Mean Absolute Error:",metrics.mean_absolute_error(y_test,y_pred))
print("mean Squared Error:" , metrics.mean_squared_error(y_test,y_pred))
```

Mean Absolute Error: 2.505418178966003
mean Squared Error: 10.18618193453022

```
In [19]: print("R2 score:",metrics.r2_score(y_test,y_pred))  
  
R2 score: 0.6763151577939721
```