

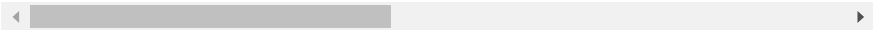
```
#importing lib
import numpy as np
from sklearn.linear_model import LinearRegression
```

```
# importing dataset
import pandas as pd
df = pd.read_csv('/content/Video_Games_Sales_as_at_22_Dec_2016.csv')
dx=df.dropna()
dx
```



| | Name | Platform | Year_of_Release | Genre | Publisher | NA_Sales | EU_Sales |
|-------|-----------------------------------|----------|-----------------|----------|------------------------------|----------|----------|
| 0 | Wii Sports | Wii | 2006.0 | Sports | Nintendo | 41.36 | 28.96 |
| 2 | Mario Kart Wii | Wii | 2008.0 | Racing | Nintendo | 15.68 | 12.76 |
| 3 | Wii Sports Resort | Wii | 2009.0 | Sports | Nintendo | 15.61 | 10.93 |
| 6 | New Super Mario Bros. | DS | 2006.0 | Platform | Nintendo | 11.28 | 9.14 |
| 7 | Wii Play | Wii | 2006.0 | Misc | Nintendo | 13.96 | 9.18 |
| ... | ... | ... | ... | ... | ... | ... | ... |
| 16667 | E.T. The Extra-Terrestrial | GBA | 2001.0 | Action | NewKidCo | 0.01 | 0.00 |
| 16677 | Mortal Kombat: Deadly Alliance | GBA | 2002.0 | Fighting | Midway Games | 0.01 | 0.00 |
| 16696 | Metal Gear Solid V: Ground Zeroes | PC | 2014.0 | Action | Konami Digital Entertainment | 0.00 | 0.01 |
| 16700 | Breach | PC | 2011.0 | Shooter | Destineer | 0.01 | 0.00 |
| 16706 | STORM: Frontline Nation | PC | 2011.0 | Strategy | Unknown | 0.00 | 0.01 |

6825 rows × 16 columns



```
dx.size
```

109200

```
dx.shape
```

(6825, 16)

```
#Separate the independent variable(s) (X) and dependent variable (y) from the dataset.
X = df['Critic_Score'].values.reshape(-1,1)
y = df['Global_Sales'].values.reshape(-1,1)
dx
```

| | Name | Platform | Year_of_Release | Genre | Publisher | NA_Sales | EU_Sales | JP_Sales | Other_S |
|-------|-----------------------------------|----------|-----------------|----------|------------------------------|----------|----------|----------|---------|
| 0 | Wii Sports | Wii | 2006.0 | Sports | Nintendo | 41.36 | 28.96 | 3.77 | |
| 2 | Mario Kart Wii | Wii | 2008.0 | Racing | Nintendo | 15.68 | 12.76 | 3.79 | |
| 3 | Wii Sports Resort | Wii | 2009.0 | Sports | Nintendo | 15.61 | 10.93 | 3.28 | |
| 6 | New Super Mario Bros. | DS | 2006.0 | Platform | Nintendo | 11.28 | 9.14 | 6.50 | |
| 7 | Wii Play | Wii | 2006.0 | Misc | Nintendo | 13.96 | 9.18 | 2.93 | |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 16667 | E.T. The Extra-Terrestrial | GBA | 2001.0 | Action | NewKidCo | 0.01 | 0.00 | 0.00 | |
| 16677 | Mortal Kombat: Deadly Alliance | GBA | 2002.0 | Fighting | Midway Games | 0.01 | 0.00 | 0.00 | |
| 16696 | Metal Gear Solid V: Ground Zeroes | PC | 2014.0 | Action | Konami Digital Entertainment | 0.00 | 0.01 | 0.00 | |
| 16700 | Breach | PC | 2011.0 | Shooter | Destineer | 0.01 | 0.00 | 0.00 | |
| 16706 | STORM: Frontline | PC | 2011.0 | Strategy | Unknown | 0.00 | 0.01 | 0.00 | |

```
#Create an instance of the LinearRegression class from scikit-learn and fit it to the data
from sklearn.linear_model import LinearRegression as lr
model = lr()
```

```
# the trained model to make predictions on new data.
lr = LinearRegression()
lr.fit(X, y)
```

LinearRegression

LinearRegression()

```
#the trained model to make predictions on new data.y_pred = lr.predict(X)
y_pred
```

```
array([[0.97023057],
       [1.17202224],
       [1.10475835],
       ...,
       [1.10475835],
       [0.46575139],
       [0.43211944]])
```

```
#Finally, evaluate the performance of the model by calculating metrics such as the R-squared value
from sklearn.metrics import r2_score

r2 = r2_score(y, y_pred)

print(f"R-squared: {r2}")
```

R-squared: 0.05643272118927545

✓ 0s completed at 2:08 PM

● ×