



```
In [8]: import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from google.colab import drive
drive.mount('/content/drive', force_remount=True)
```

Mounted at /content/drive

```
In [ ]: import glob
files=glob.glob('/content/drive/My Drive/**/*.csv', recursive=True)
for file in files:
    print(file)
```

/content/drive/My Drive/MACHINE LEARNING/IRIS - Copy.csv
/content/drive/My Drive/MACHINE LEARNING/Colab Notebooks/headbrain.csv
/content/drive/My Drive/MACHINE LEARNING/Colab Notebooks/suv_data.csv

```
In [3]: file_path='/content/drive/My Drive/MACHINE LEARNING/Colab Notebooks/headbrain.csv'
pd.read_csv(file_path)
```

Out[3]:

	Gender	Age Range	Head Size(cm^3)	Brain Weight(grams)
0	1	1	4512	1530
1	1	1	3738	1297
2	1	1	4261	1335
3	1	1	3777	1282
4	1	1	4177	1590
...
232	2	2	3214	1110
233	2	2	3394	1215
234	2	2	3233	1104
235	2	2	3352	1170
236	2	2	3391	1120

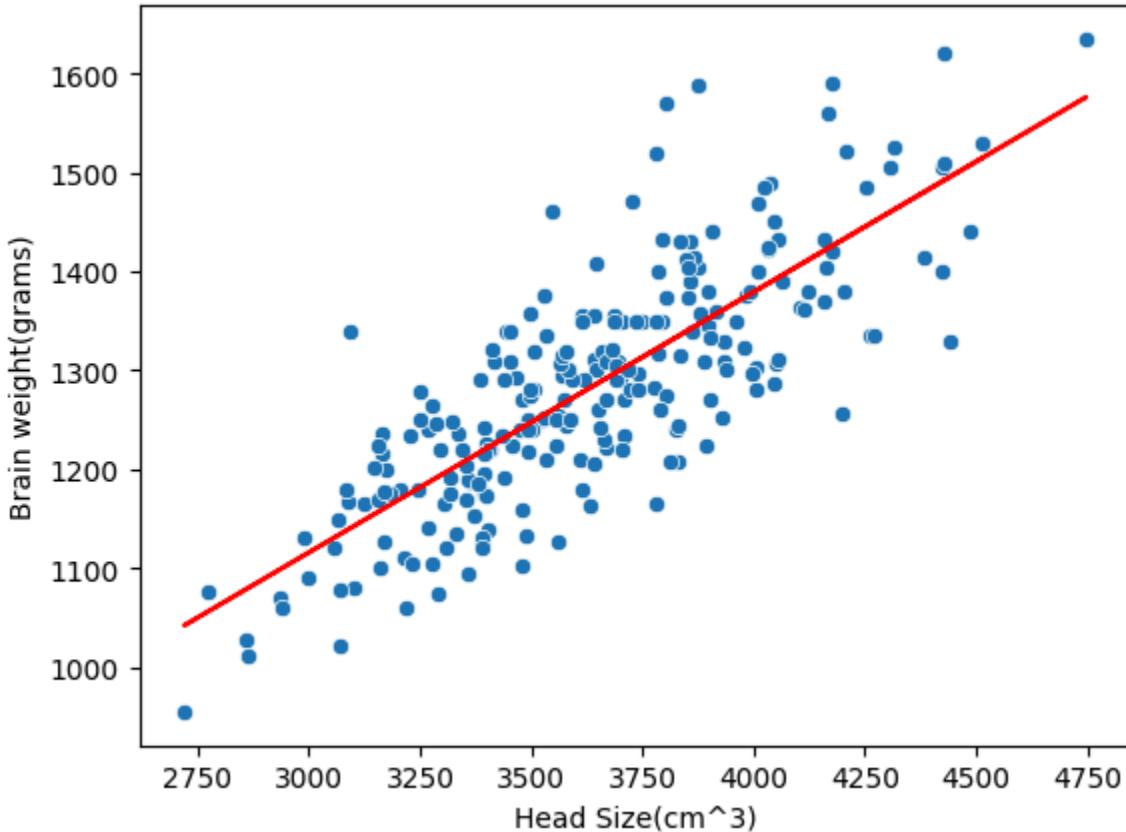
237 rows × 4 columns

```
In [4]: df = pd.read_csv(file_path)
X, Y = np.array(df['Head Size(cm^3)']), np.array(df['Brain Weight(grams)'])
mean_X = np.mean(X)
mean_Y = np.mean(Y)
mean_XY = np.mean(X * Y)
mean_X2 = np.mean(X * X)
```

```
In [5]: w1=(mean_XY - mean_X*mean_Y)/(mean_X2-mean_X**2)
```

```
w0=mean_Y-w1*mean_X  
y_pred=w0+w1*X  
sns.scatterplot(x=X,y=Y)  
plt.xlabel("Head Size(cm^3)")  
plt.ylabel("Brain weight(grams)")  
plt.plot(X,y_pred,color="red")
```

Out[5]: [<matplotlib.lines.Line2D at 0x7aaa88558d40>]



In [6]: `print(np.mean((Y-y_pred)**2))`

5201.3840280023305

```
from sklearn.metrics import mean_squared_error  
from sklearn.linear_model import LinearRegression  
  
X=X.reshape(-1,1)  
model=LinearRegression()  
model.fit(X,Y)  
w1=model.coef_[0]  
w0= model.intercept_  
y_model_pred=model.predict(X)  
mse_model= mean_squared_error(Y,y_model_pred)  
print("Mean Squared Error:",mse_model)  
  
plt.scatter(X,Y)  
plt.plot(X,y_model_pred,color="red")  
plt.xlabel("Head Size(cm^3)")
```

```
plt.ylabel("Brain weight(grams)")
```

Mean Squared Error: 5201.3840280023305

Out[7]: Text(0, 0.5, 'Brain weight(grams)')

