Course Name: Business Intelligence Lab **Course Code:** CSP-421

Experiment-1.3

Aim:

Write a program to implement Sales Revenue Prediction using Linear Regression.

Software Required:

Any language compiler Python, Jupyter Notebook , Google Colab, Dataset.

Description:

The experiment "Implementation of Sales Revenue Prediction using Linear Regression" focuses on developing a program that utilizes linear regression to predict sales revenue based on historical data. The experiment involves understanding the concept of linear regression, preparing the dataset, implementing the linear regression algorithm, evaluating the model's performance, and using it for sales revenue prediction.

Pseudo code/Algorithms/Flowchart/Steps:

1. **Dataset Preparation:**

Introduce the dataset containing historical sales revenue and independent variables. Load the dataset into the programming environment. Perform necessary data preprocessing steps, such as handling missing values, scaling, or encoding categorical variables.

2. **Exploratory Data Analysis:**

Perform exploratory data analysis to gain insights into the dataset. Analyse the relationships between the independent variables and the dependent variable (sales revenue) using visualizations, correlation analysis, or summary statistics.

3. **Programming Environment Setup:**

Set up the programming environment with the chosen programming language. Import any necessary libraries or packages for data manipulation, visualization, and linear regression.

4. **Implementing Linear Regression:**

Write a program to implement linear regression for sales revenue prediction. Divide the dataset into training and testing sets for model evaluation.Fit a linear regression model to the training data using the independent

variables and the corresponding sales revenue. Validate the model using the testing data and calculate relevant evaluation metrics such as mean squared error (MSE) or R-squared.

5. **Model Evaluation and Interpretation:**

Evaluate the performance of the linear regression model based on the evaluation metrics obtained. Interpret the coefficients of the independent variables in the linear regression equation to understand their impact on sales revenue. Discuss any assumptions and limitations of the linear regression model.

6. **Sales Revenue Prediction:**

Utilize the trained linear regression model to predict sales revenue for new or unseen data points. Input relevant independent variables (e.g., advertising expenditure, market size) into the model and obtain the predicted sales revenue.

7. **Experimentation and Analysis:**

Implementation and Output:

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import sklearn
import warnings
warnings.filterwarnings('ignore')

df=pd.read_csv('advertising.csv')
df.head(6)
```

	TV	Radio	Newspaper	Sales
0	230.1	37.8	69.2	22.1
1	44.5	39.3	45.1	10.4
2	17.2	45.9	69.3	12.0
3	151.5	41.3	58.5	16.5
4	180.8	10.8	58.4	17.9
5	8.7	48.9	75.0	7.2

Course Name: Business Intelligence Lab **Course Code:** CSP-421

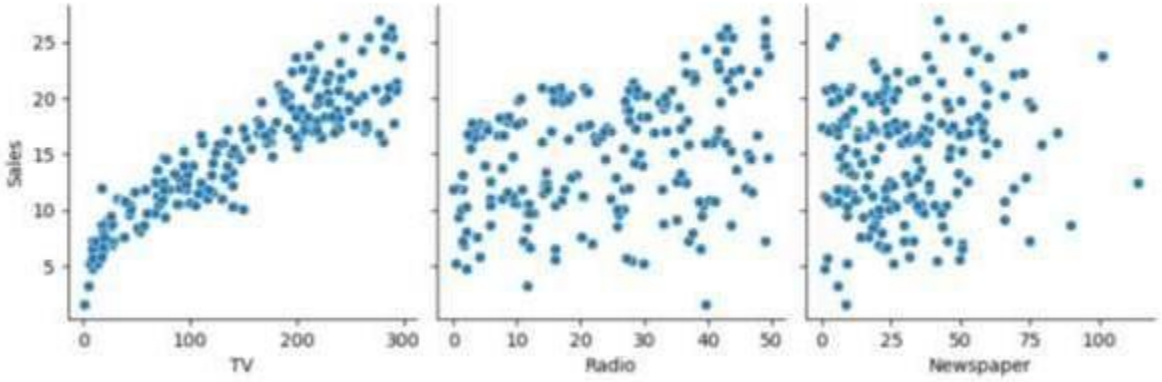
```
[2] pd.DataFrame(df.isnull().sum(),columns=['Count of null values']).T
```

	TV	Radio	Newspaper	Sales
Count of null values	0	0	0	0

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 200 entries, 0 to 199
Data columns (total 4 columns):
 #   Column     Non-Null Count  Dtype
---  ------     --------------  -----
 0   TV         200 non-null    float64
 1   Radio      200 non-null    float64
 2   Newspaper  200 non-null    float64
 3   Sales      200 non-null    float64
dtypes: float64(4)
memory usage: 6.4 KB
```

```
[6] sns.pairplot(df, x_vars=['TV','Radio','Newspaper'],y_vars=['Sales'],height=3,aspect=1)
    plt.show()
```



:

Course Name: Business Intelligence Lab **Course Code:** CSP-421

```
df.std()
```

```
TV           85.854236
Radio        14.846809
Newspaper    21.778621
Sales         5.283892
dtype: float64
```

```
[8] df.corr()
```

	TV	Radio	Newspaper	Sales
TV	1.000000	0.054809	0.056648	0.901208
Radio	0.054809	1.000000	0.354104	0.349631
Newspaper	0.056648	0.354104	1.000000	0.157960
Sales	0.901208	0.349631	0.157960	1.000000

.

```
[9] df.var()
```

```
TV           7370.949893
Radio         220.427743
Newspaper     474.308326
Sales          27.919517
dtype: float64
```

```
df.mean()
```

```
TV           147.0425
Radio         23.2640
Newspaper     30.5540
Sales         15.1305
dtype: float64
```

```
[19] print("The LR model is: Y=",lr.intercept_, "+",lr.coef_,"radio ")
```

```
The LR model is: Y= 6.889929307794299 + [0.05671244] radio
```

```
[20] lr.score(X_train,Y_train)
```

```
0.822322146620674
```

```
lr.score(X_test,Y_test)
```

```
0.7281236097879917
```

Course Name: Business Intelligence Lab **Course Code:** CSP-421

```
[22] Y_pred=lr.predict(X_test)
     Y_pred
```

```
array([16.85430492, 20.18899637,  8.23968537, 14.22851898, 12.34566599,
       19.27592609, 20.05288651,  7.37765629, 10.82036551, 11.89763771,
       20.48957229, 16.9280311 , 19.18518619, 16.67282512, 21.79962965,
       14.55745113, 13.75213448, 19.83737924,  9.32856421, 14.35895759,
       13.11695516, 11.160376  , 10.42311429, 19.50277585, 22.41212399,
       10.76906017, 12.01673384, 12.31730977, 17.37605937, 18.61806179,
       13.87123061, 15.38545274,  7.84836953, 10.00630735, 11.22275968,
        9.42497535, 12.71429684, 21.78828716, 11.83525403, 11.22275968])
```

```
diff = pd.DataFrame({'Actual':Y_test, 'Predicted':Y_pred})
diff.head(5)
```

	Actual	Predicted
112	17.1	16.854305
165	16.9	20.188996
12	9.2	8.239685
73	11.0	14.228519
144	12.3	12.345686

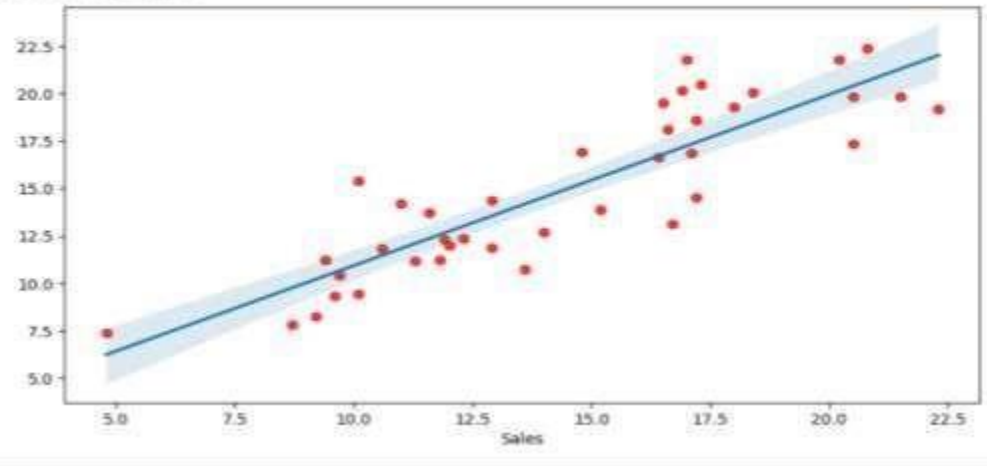
```
from sklearn import metrics
from sklearn.metrics import r2_score
R2=r2_score(Y_test,Y_pred)
mae=metrics.mean_absolute_error(Y_test, Y_pred)
mse=metrics.mean_squared_error(Y_test,Y_pred)
rmse=np.sqrt(metrics.mean_squared_error(Y_test,Y_pred))
print('Accuracy =',R2.round(2)*100,"%")
print('mae =',mae.round(2))
print('mse =',mse.round(2))
print('rmse=', rmse.round(2))
```

```
Accuracy = 71.0 %
mae = 1.74
mse = 4.66
rmse= 2.16
```

Course Name: Business Intelligence Lab **Course Code:** CSP-421

```
plt.figure(figsize=(10,5))
sns.regplot(x=Y_test, y=Y_pred, scatter_kws={'color': 'red'})
```

```
<Axes: xlabel='Sales'>
```



Learning Outcomes:

1. Understand the concepts and principles of linear regression as a predictive modeling technique.
2. Gain proficiency in programming and implementing linear regression algorithms.
3. Apply data preprocessing techniques for preparing the dataset for regression analysis.
4. Perform feature selection and engineering to enhance the predictive power of the model.
5. Assess and interpret the performance of the linear regression model.