EXPERIMENT NO. 1

AIM: Study and Implement Linear Regression.

Theory:

Linear Regression is a fundamental machine learning algorithm used for predicting a continuous target variable based on one or more input features (also called independent variables). The idea behind linear regression is to establish a linear relationship between the target variable (dependent variable) and the input features.

In simple terms, Linear Regression assumes that the target variable can be represented as a linear combination of the input variables:

$$\mathbf{Y = m\,X + C}$$

Where:

- Y is the dependent variable (target/predicted value).
- C is the intercept (the value of y when x=0).
- m is the coefficient (the change in y with a unit change in x).
- X is the independent variable (input feature).

The goal of linear regression is to find the best-fitting line that minimizes the differences between the actual and predicted values of the target variable.

Types of Linear Regression

1. **Simple Linear Regression**: Involves only one input feature.
2. **Multiple Linear Regression**: Involves two or more input features.

The implementation discussed in this theory is based on **Simple Linear Regression**.

Key Concepts in Linear Regression

1. Line of Best Fit

Linear regression tries to find the line that best fits the data. This is done by minimizing the sum of squared differences between actual and predicted values, known as **least squares**.

2. Model Parameters (Intercept and Coefficient)

The **coefficient** determines the slope of the line, which indicates how much the target variable changes with a unit change in the input variable. The **intercept** is the value of the target variable when the input variable is zero.

3. Making Predictions

Once the model is trained, it can be used to make predictions for new data. The predicted values are based on the linear equation formed during training.

4. Evaluation

For simple linear regression, common evaluation metrics include:

- **Mean Squared Error (MSE)**: Measures the average squared difference between actual and predicted values.
- **R-squared**: Indicates the proportion of variance in the target variable explained by the model.

Code:

```
# In[1]:
#Importing the libraries
import pandas as pd
import numpy as np
from sklearn import linear_model
import matplotlib.pyplot as plt

# In[2]:
#import csv files
data = pd.read_csv('train.csv')
data.head()

# In[3]:
#Visualizing the data using scatter plot
plt.xlabel('Above Ground Living Area (sq. ft.)')
plt.ylabel('Sale Price (in US $)')
plt.scatter(data['LotArea'], data['SalePrice'], color='red', marker='+')
plt.show()

#In[4]:
data_cleaned = data[['LotArea', 'SalePrice']].dropna()

#In[5]:
reg = linear_model.LinearRegression()
reg.fit(data_cleaned[['LotArea']], data_cleaned['SalePrice'])

#In[6]:
predicted_price = reg.predict([[3300]])
print(f'Predicted price for a house with 3300 sq. ft. living area: {predicted_price}')
```

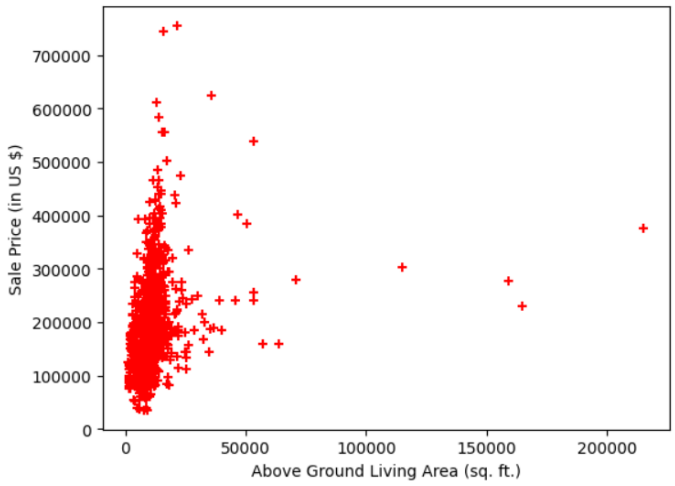
```
#In[7]:
print("Intercept:", reg.intercept_)
print("Coefficient:", reg.coef_)

#In[8]:
new_areas = pd.DataFrame({'LotArea': [2000, 2500, 3000, 3500, 4000]})
new_areas['PredictedPrice'] = reg.predict(new_areas[['LotArea']])
print(new_areas)

#In[9]:
new_areas.to_csv("house_price_predictions_kaggle.csv", index=False)
```

Output Snapshots:

	Id	MSSubClass	MSZoning	LotFrontage	LotArea	Street	Alley	LotShape	LandContour	Utilities	...	PoolArea	PoolQC	Fence	MiscFeature	MiscVal	MoSold
0	1	60	RL	65.0	8450	Pave	NaN	Reg	Lvl	AllPub	...	0	NaN	NaN	NaN	0	2
1	2	20	RL	80.0	9600	Pave	NaN	Reg	Lvl	AllPub	...	0	NaN	NaN	NaN	0	5
2	3	60	RL	68.0	11250	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	NaN	NaN	0	9
3	4	70	RL	60.0	9550	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	NaN	NaN	0	2
4	5	60	RL	84.0	14260	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	NaN	NaN	0	12



LinearRegression

LinearRegression()

```
Predicted price for a house with 3300 sq. ft. living area: [165766.05933751]
E:\Python\Lib\site-packages\sklearn\base.py:493: UserWarning: X does not have valid feature names, but LinearRegression was fitted with feature names
  warnings.warn(
```

```
Intercept: 158836.1518968766
Coefficient: [2.09997195]
```

	LotArea	PredictedPrice
0	2000	163036.095800
1	2500	164086.081776
2	3000	165136.067752
3	3500	166186.053728
4	4000	167236.039704

Learning Outcome: