

Implement multiple linear regression for the data sets 'company_data.csv'

I. Evaluate the model by finding

a. Mean Squared Error

b. R-Squared

II. Display scatterplot for the data

III. Find model coefficients and intercept

```
# Import necessary libraries
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score

# Load the dataset
advertising = pd.read_csv('company_data.csv')

# Display the first few rows of the data
print(advertising.head())

# Scatter plot representation of data using seaborn
import matplotlib.pyplot as plt
import seaborn as sns
sns.pairplot(advertising, x_vars=['TV', 'Radio', 'Newspaper'],
             y_vars='Sales', height=5, aspect=1, kind='scatter')
plt.show()

# Assuming 'Sales' is the dependent variable and the rest are features
X = advertising.drop(columns=['Sales'])
y = advertising['Sales']

# Split the dataset into training and testing sets (80% training, 20%
testing)
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2)

# Create the linear regression model
model = LinearRegression()
```

```

# Train the model on the training data
model.fit(X_train, y_train)

# Predict on the test data
y_pred = model.predict(X_test)

# Model evaluation
mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)

# Display results
print("\nMean Squared Error:", mse)
print("R-squared:", r2)

# Display the coefficients and intercept
print("\nCoefficients:", model.coef_)
print("Intercept:", model.intercept_)

#To Display Real Values and Predicted Values
y_pred = model.predict(X_test)
for(i,j) in zip(y_test,y_pred):
    if i!=j:
        print("Actual value :",i,"Predicted value :",j)
print("\nNumber of mislabeled points from test data set :", (y_test !=
y_pred).sum())

```