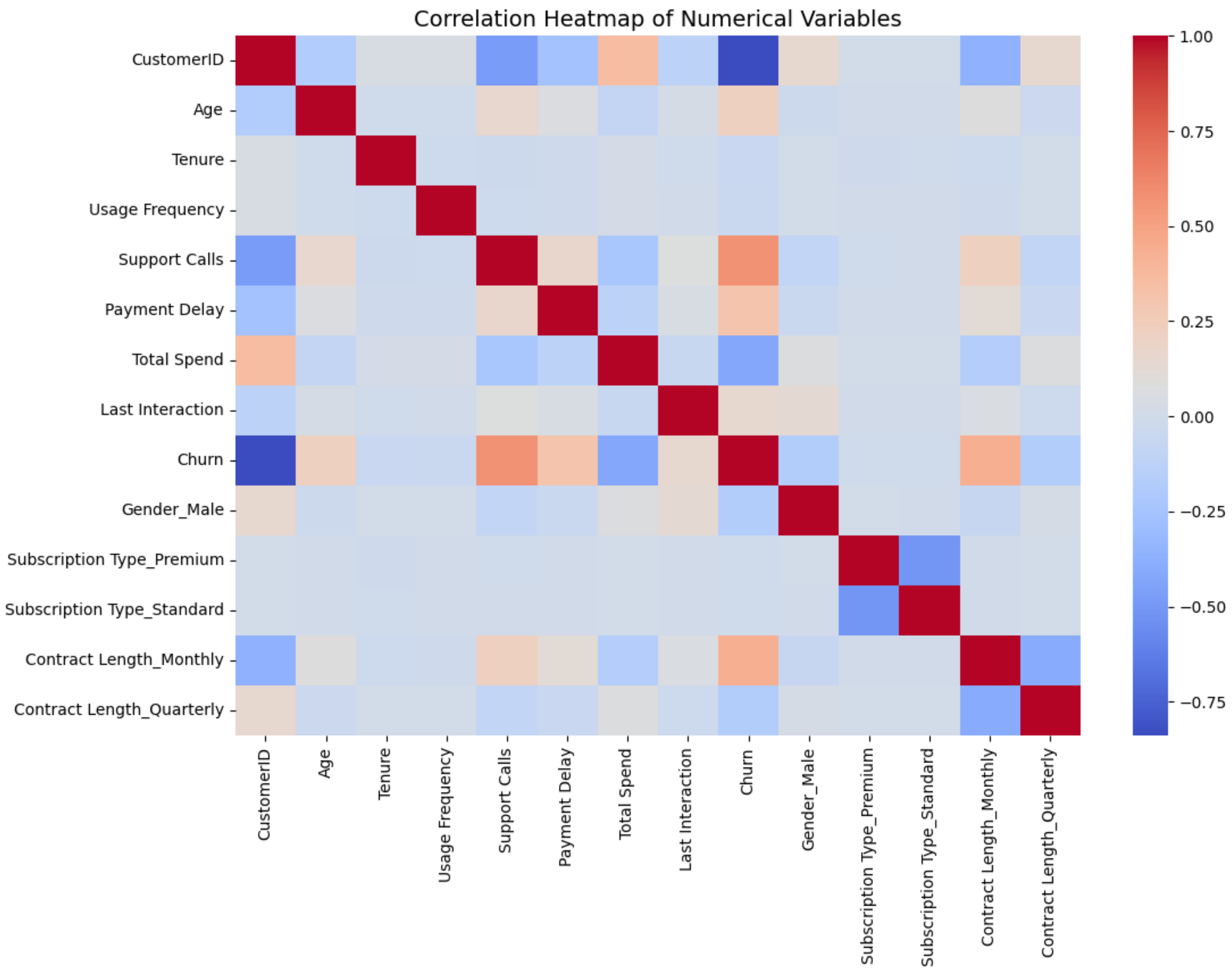
```
from google.colab import drive
drive.mount('/content/drive')
```

1)Implement Simple Linear regression on a given dataset. 2)Draw all graphs and sub,it a pdf with all results. 3)Also on the answersheet explain the dataset and findings of simple linear regression on the dataset chose.

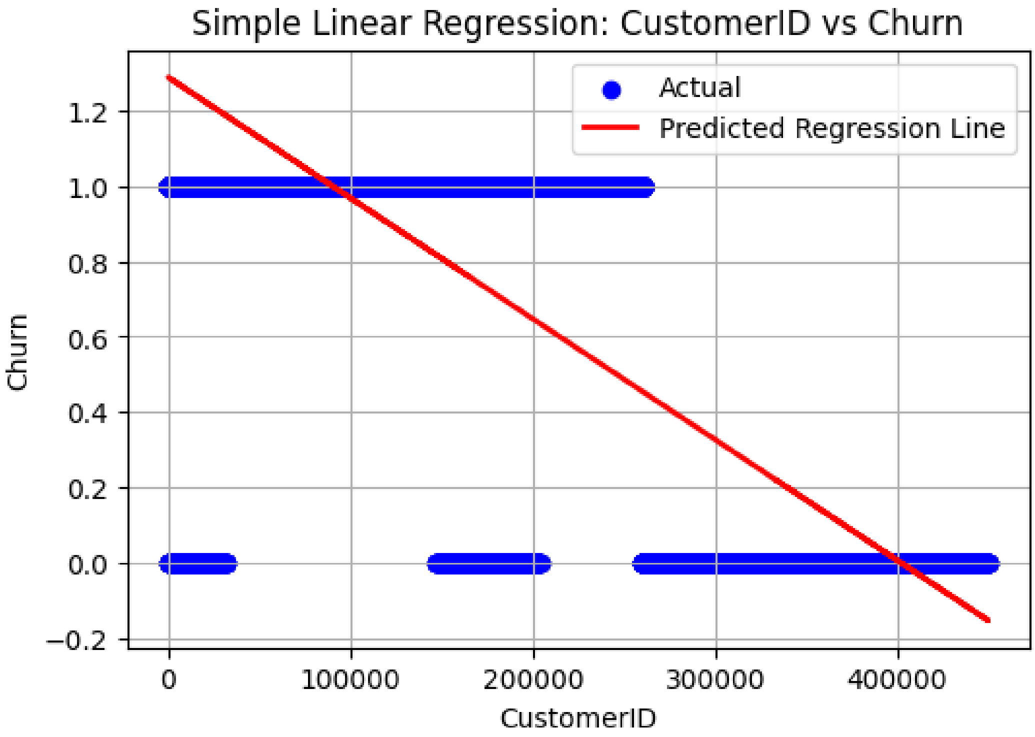
```
# ============================================
# □ SIMPLE LINEAR REGRESSION - CUSTOMER CHURN DATA
# ============================================

# STEP 1: Import Libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import r2_score, mean_squared_error

# STEP 2: Load Dataset
from google.colab import files
uploaded = files.upload()  # Upload your file manually in Colab

# Replace with your uploaded filename
df = pd.read_csv('/content/customer_churn_dataset-training-
master[1].csv')

# STEP 3: Inspect Dataset
print("Shape of dataset:", df.shape)
print(df.head())

# STEP 4: Handle Missing Values
df = df.dropna()

# STEP 5: Convert Categorical Columns (if any) to Numeric
df_encoded = pd.get_dummies(df, drop_first=True)

# STEP 6: Correlation Heatmap
plt.figure(figsize=(12,8))
sns.heatmap(df_encoded.corr(), cmap='coolwarm', annot=False)
plt.title("Correlation Heatmap of Numerical Variables", fontsize=14)
plt.show()

# ============================================
# STEP 7: SIMPLE LINEAR REGRESSION
# We'll pick 5 numeric columns as X and use one target (say 'Churn' or
similar)
```

```
# ============================================

# Choose target column – adjust this if your target is named
differently
target_col = 'Churn' if 'Churn' in df_encoded.columns else
df_encoded.columns[-1]
print(f"Using target variable: {target_col}")

# Select 5 numeric predictor columns (excluding target)
numeric_cols =
df_encoded.select_dtypes(include=[np.number]).columns.tolist()
numeric_cols = [col for col in numeric_cols if col != target_col]
chosen_features = numeric_cols[:5]  # first 5 features for demo

print("Selected features for SLR:", chosen_features)

# STEP 8: Apply SLR and Plot
for feature in chosen_features:
    X = df_encoded[[feature]]
    y = df_encoded[target_col]

    X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.2, random_state=42)

    model = LinearRegression()
    model.fit(X_train, y_train)
    y_pred = model.predict(X_test)

    # Metrics
    r2 = r2_score(y_test, y_pred)
    mse = mean_squared_error(y_test, y_pred)

    print(f"\nFeature: {feature}")
    print(f"R² Score: {r2:.4f}")
    print(f"MSE: {mse:.4f}")
    print(f"Intercept: {model.intercept_:.4f}, Coefficient:
{model.coef_[0]:.4f}")

    # Plot Regression Line
    plt.figure(figsize=(6,4))
    plt.scatter(X_test, y_test, color='blue', label='Actual')
    plt.plot(X_test, y_pred, color='red', linewidth=2,
label='Predicted Regression Line')
    plt.title(f'Simple Linear Regression: {feature} vs {target_col}')
    plt.xlabel(feature)
    plt.ylabel(target_col)
    plt.legend()
    plt.grid(True)
    plt.show()
```

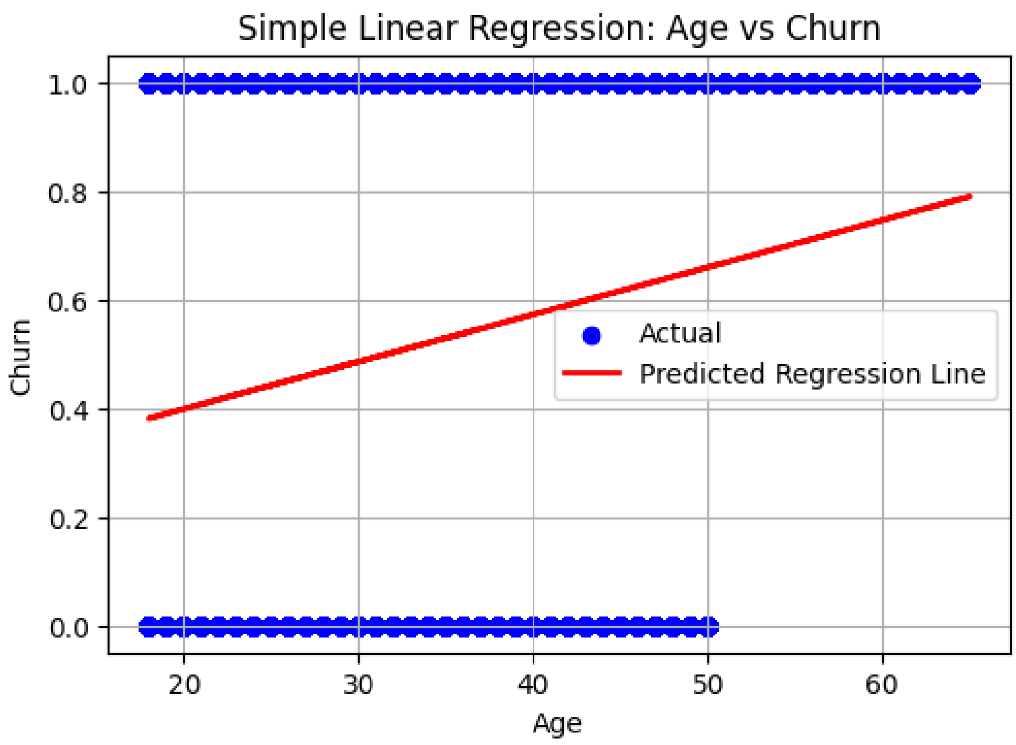
```
<IPython.core.display.HTML object>

Saving customer_churn_dataset-training-master[1].csv to
customer_churn_dataset-training-master[1] (1).csv
Shape of dataset: (440833, 12)
   CustomerID   Age  Gender  Tenure  Usage Frequency  Support Calls  \
0         2.0  30.0  Female    39.0             14.0            5.0
1         3.0  65.0  Female    49.0              1.0           10.0
2         4.0  55.0  Female    14.0              4.0            6.0
3         5.0  58.0    Male    38.0             21.0            7.0
4         6.0  23.0    Male    32.0             20.0            5.0

   Payment Delay Subscription Type Contract Length  Total Spend  \
0           18.0          Standard          Annual        932.0
1            8.0             Basic         Monthly        557.0
2           18.0             Basic       Quarterly        185.0
3            7.0          Standard         Monthly        396.0
4            8.0             Basic         Monthly        617.0

   Last Interaction  Churn
0              17.0    1.0
1               6.0    1.0
2               3.0    1.0
3              29.0    1.0
4              20.0    1.0
```



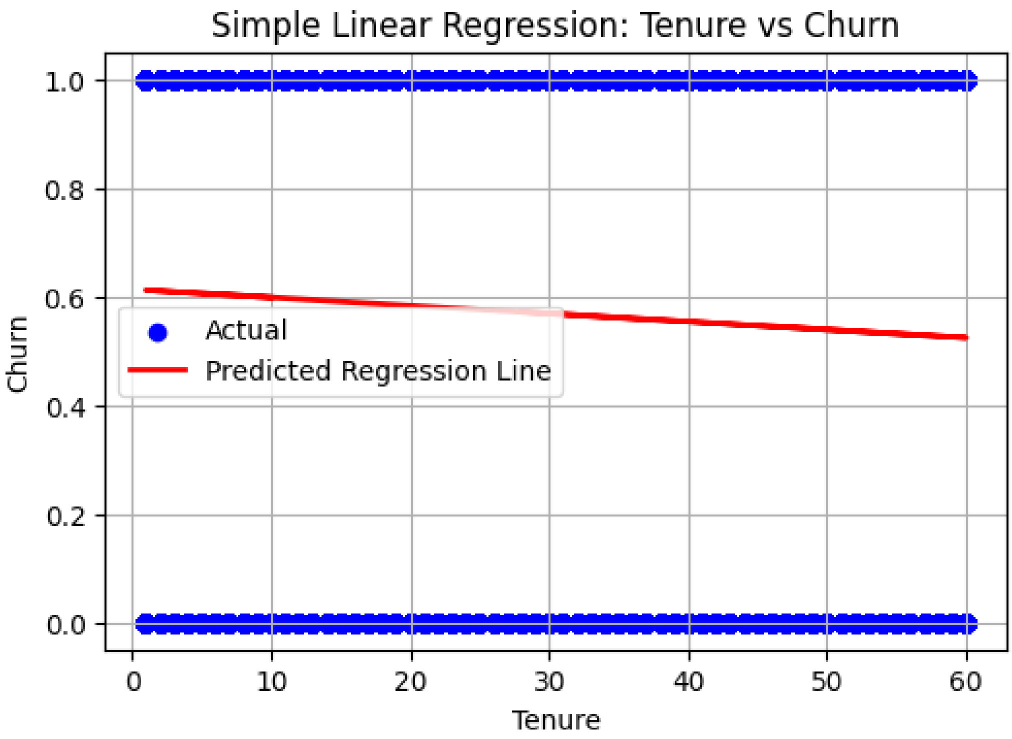
```
Using target variable: Churn
Selected features for SLR: ['CustomerID', 'Age', 'Tenure', 'Usage
Frequency', 'Support Calls']

Feature: CustomerID
R² Score: 0.7048
MSE: 0.0724
Intercept: 1.2903, Coefficient: -0.0000
```



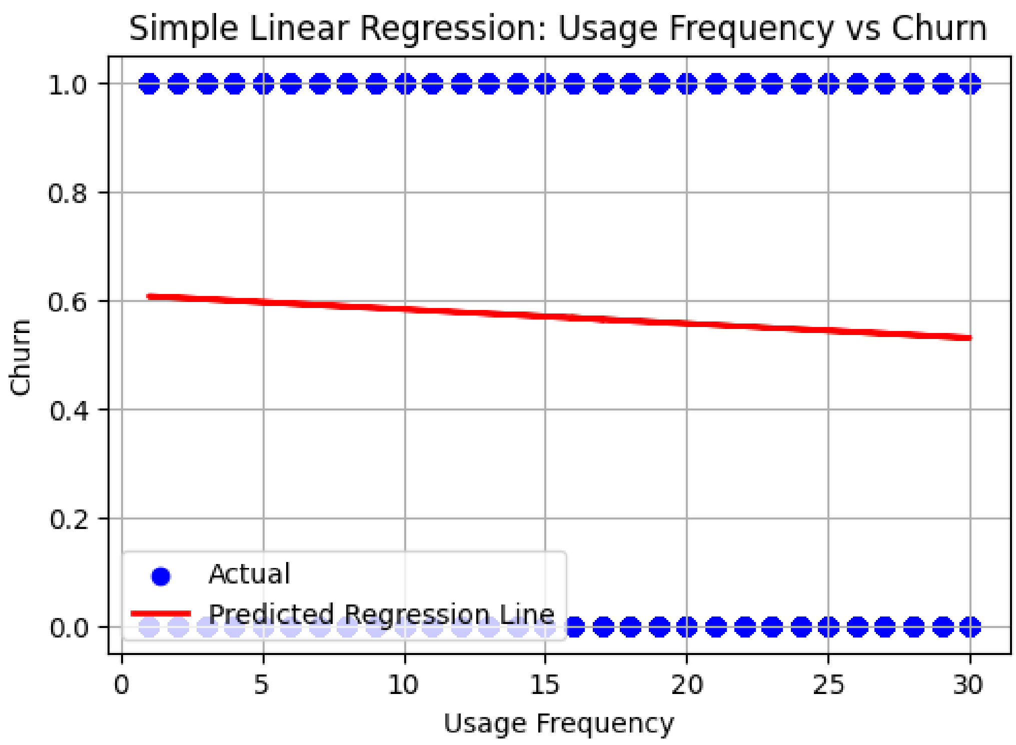
```
Feature: Age
R² Score: 0.0482
MSE: 0.2335
Intercept: 0.2249, Coefficient: 0.0087
```



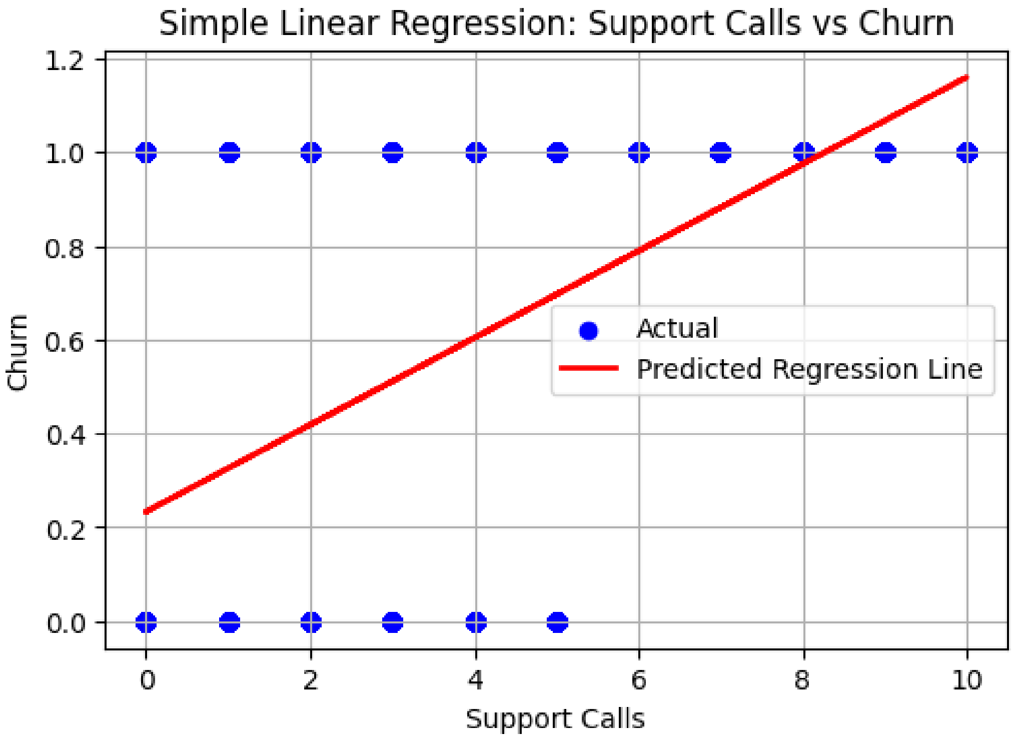
```
Feature: Tenure
R² Score: 0.0028
MSE: 0.2447
Intercept: 0.6133, Coefficient: -0.0015
```



```
Feature: Usage Frequency
R² Score: 0.0023
MSE: 0.2448
Intercept: 0.6084, Coefficient: -0.0026
```



```
Feature: Support Calls
R² Score: 0.3294
MSE: 0.1645
Intercept: 0.2329, Coefficient: 0.0927
```


Simple Linear Regression: Support Calls vs Churn
Churn
1.2
1.0
0.8
0.6
0.4
0.2
0.0
0
2
4
6
8
10
Support Calls
Actual
Predicted Regression Line