**CUSTOMER SEGMENTATION USING DATA SCIENCE**

**Phase 3:** Development Part 1

We are transitioning the project from the inventive to the developmental phase by executing various analyses such as supervised learning algorithm - Regression line, Customer Lifetime Value Analysis, Descriptive Statistics Analysis, Age based analysis, and so on.

**1)dropna():**

missing\_values = df.isna().sum()

print("Missing Values:")

print(missing\_values)

df.dropna(inplace=True)

df['Age'].fillna(df['Age'].mean(), inplace=True)

df['Annual Income (k$)'].fillna(df['Annual Income (k$)'].mean(), inplace=True)

**Output:**

Missing Values:

|  |  |
| --- | --- |
| **CustomerID** | 0 |
| **Genre** | 0 |
| **Age** | 0 |
| **Annual Income (k$)** | 0 |
| **Spending Score (1-100)** | 0 |

dtype: int64

**Note:** All of the attributes in the dataset mall\_customer.csv output is set to 0 because there are no missing values.

**2)Customer Lifetime Value (CLV) Analysis:**

clv = df['Spending Score (1-100)'].sum()

print(f"Total CLV for all customers: ${clv}")

**Output:**

Total CLV for all customers: $10040

**3)Regression Line:**

import NumPy as np

import pandas as pd

from sklearn.linear\_model import LinearRegression

import matplotlib.pyplot as plt

df = pd.read\_csv("C:/nm\_project/Mall\_Customers.csv")

print(df)

X = df[['Age', 'Annual Income (k$)']]

y = df['Spending Score (1-100)']

model = LinearRegression()

model.fit(X, y)

slope\_age, slope\_income = model.coef\_

intercept = model.intercept\_

plt.scatter(df['Age'], df['Annual Income (k$)'], c=df['Spending Score (1-100)'], cmap='viridis')

plt.xlabel('Age')

plt.ylabel('Annual Income (k$)')

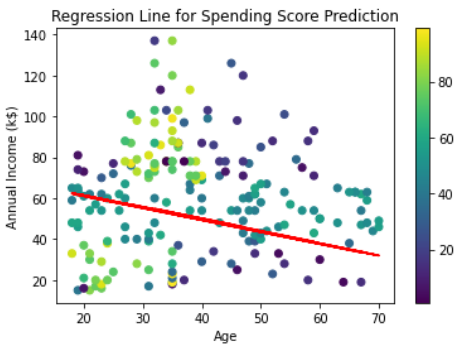
plt.title('Regression Line for Spending Score Prediction')

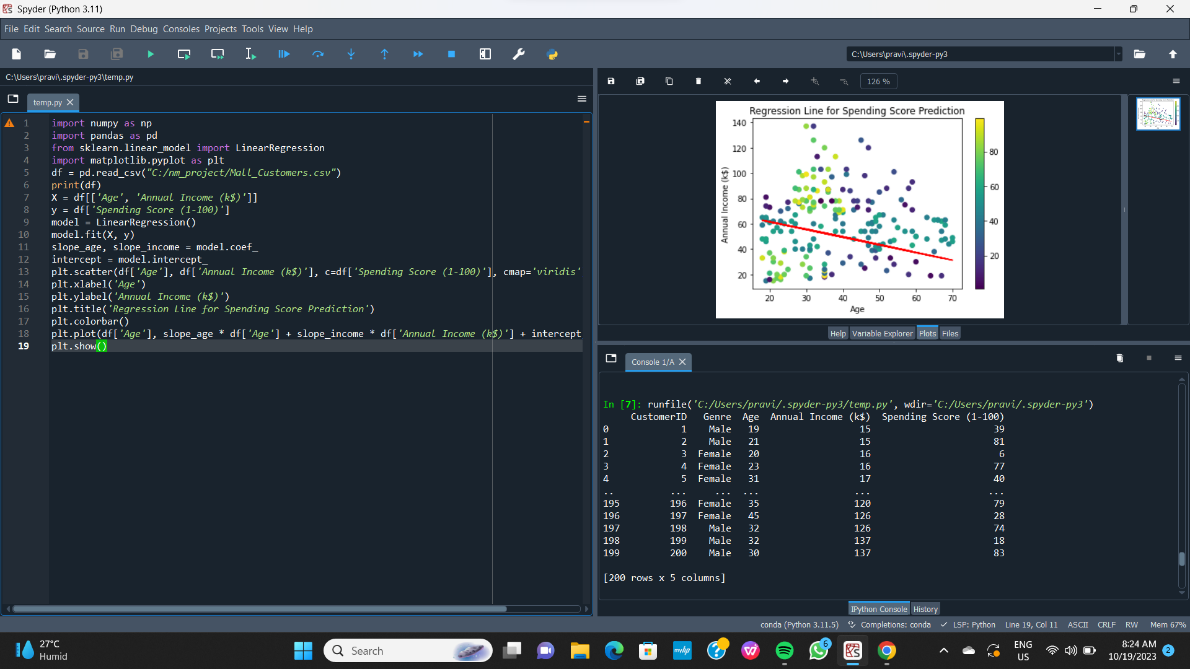
plt.colorbar()

plt.plot(df['Age'], slope\_age \* df['Age'] + slope\_income \* df['Annual Income (k$)'] + intercept, color='red')

plt.show()

**Output:**





**4)Age based analysis:**

age\_bins = [0, 20, 30, 40, 50, 60, 70, 120]

age\_labels = ["0-20", "21-30", "31-40", "41-50", "51-60", "61-70", "71+"]

df['Age Group'] = pd.cut(df['Age'], bins=age\_bins, labels=age\_labels)

age\_group\_spending = df.groupby('Age Group')['Spending Score (1-100)'].mean()

age\_group\_spending.plot(kind='bar')

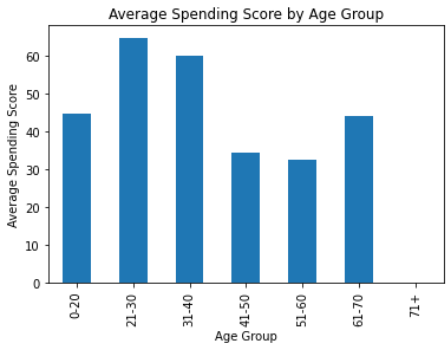
plt.title("Average Spending Score by Age Group")

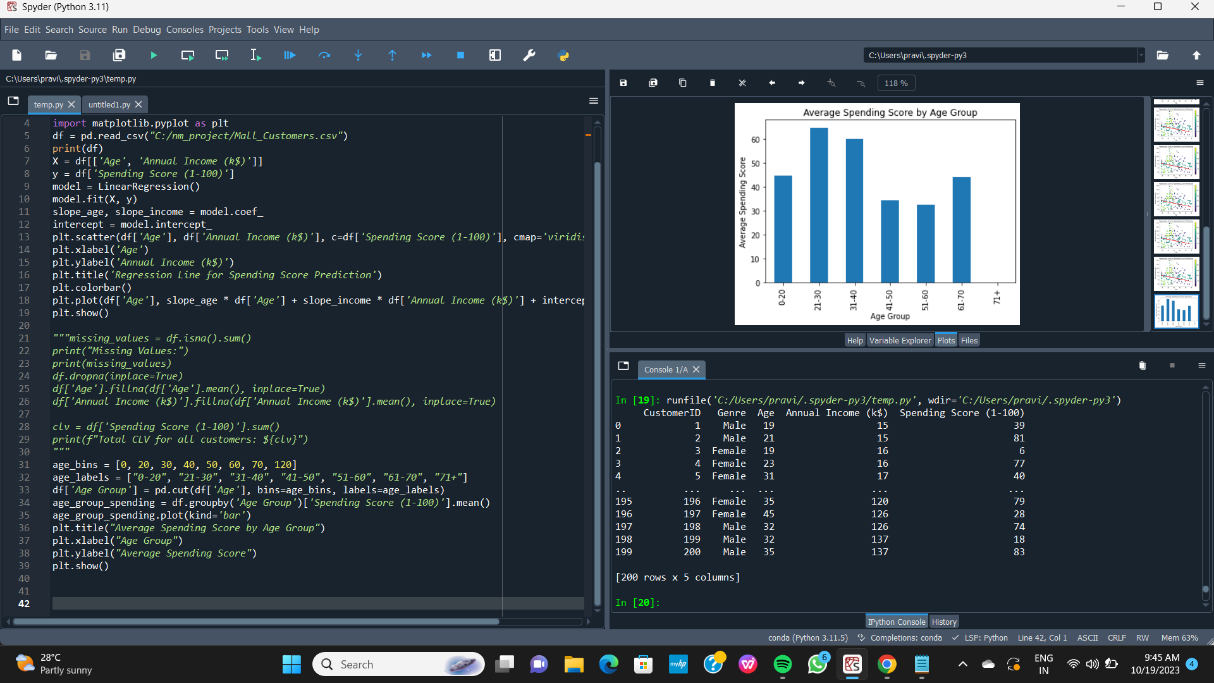
plt.xlabel("Age Group")

plt.ylabel("Average Spending Score")

plt.show()

**Output:**





**5)Descriptive Statistics Analysis:**

genre\_count = df['Genre'].value\_counts()

numerical\_stats = df[['Age', 'Annual Income (k$)', 'Spending Score (1-100)']].describe()

categorical\_stats = df[['CustomerID', 'Genre']].describe()

print("\nGenre Frequency Count:")

print(genre\_count)

print("Descriptive Statistics for Numerical Columns:")

print(numerical\_stats)

print("\nDescriptive Statistics for Categorical Columns:")

print(categorical\_stats)

**Output:**

**Genre Frequency Count:**

|  |  |
| --- | --- |
| **Genre** |  |
| Female | 112 |
| Male | 88 |

Name: count, dtype: int64

**Descriptive Statistics for Numerical Columns:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Age** | **Annual Income (k$)** | **Spending Score (1-100)** |
| **count** | 200.000000 | 200.000000 | 200.000000 |
| **mean** | 39.000000 | 60.560000 | 50.200000 |
| **std** | 13.922975 | 26.264721 | 25.823522 |
| **min** | 18.000000 | 15.000000 | 1.000000 |
| **25%** | 28.750000 | 41.500000 | 34.750000 |
| **50%** | 36.000000 | 61.500000 | 50.000000 |
| **75%** | 49.000000 | 78.000000 | 73.000000 |
| **max** | 70.000000 | 137.000000 | 99.000000 |

**Descriptive Statistics for Categorical Columns:**

|  |  |
| --- | --- |
|  | **CustomerID** |
| **count** | 200.000000 |
| **mean** | 100.500000 |
| **std** | 57.879185 |
| **min** | 1.000000 |
| **25%** | 50.750000 |
| **50%** | 100.500000 |
| **75%** | 150.250000 |
| **max** | 200.000000 |