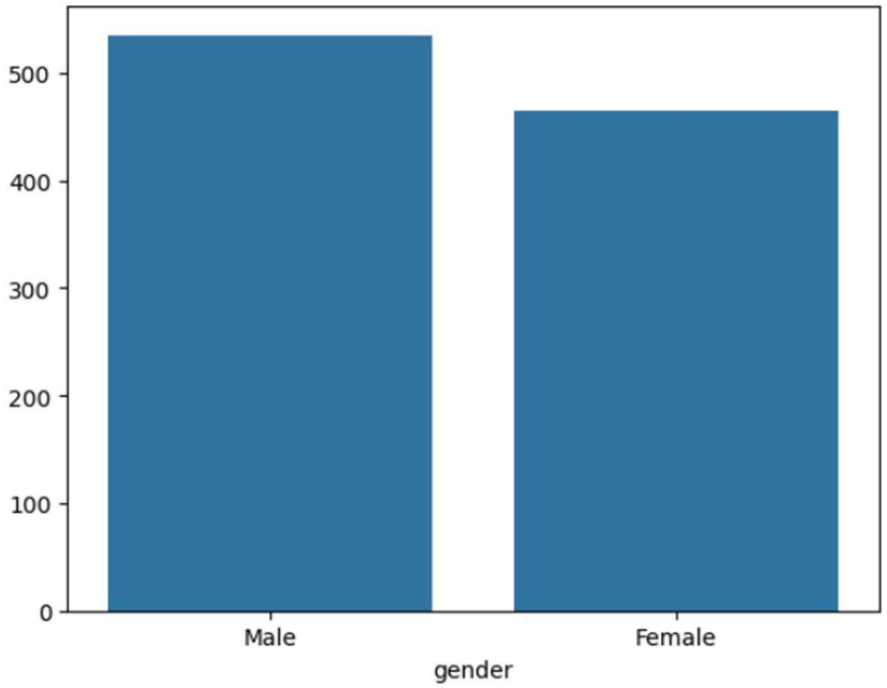
OUTPUT SCREENSHOTS

Gender Count

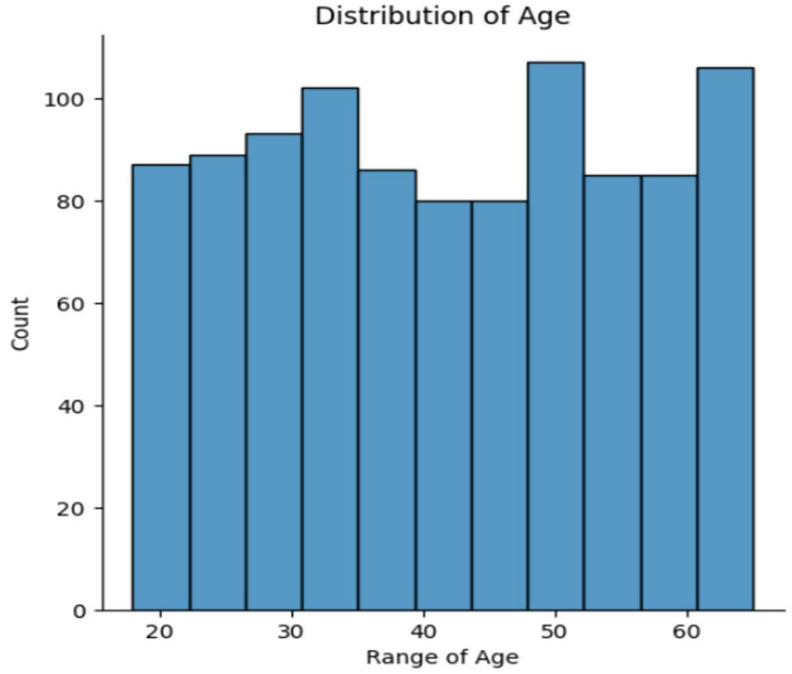
```
genders = data.gender.value_counts()
sns.barplot(x=genders.index, y=genders.values)
plt.show()
```



Distribution of Age

```
sns.displot(data['age'])
plt.title('Distribution of Age')
plt.xlabel('Range of Age')
```

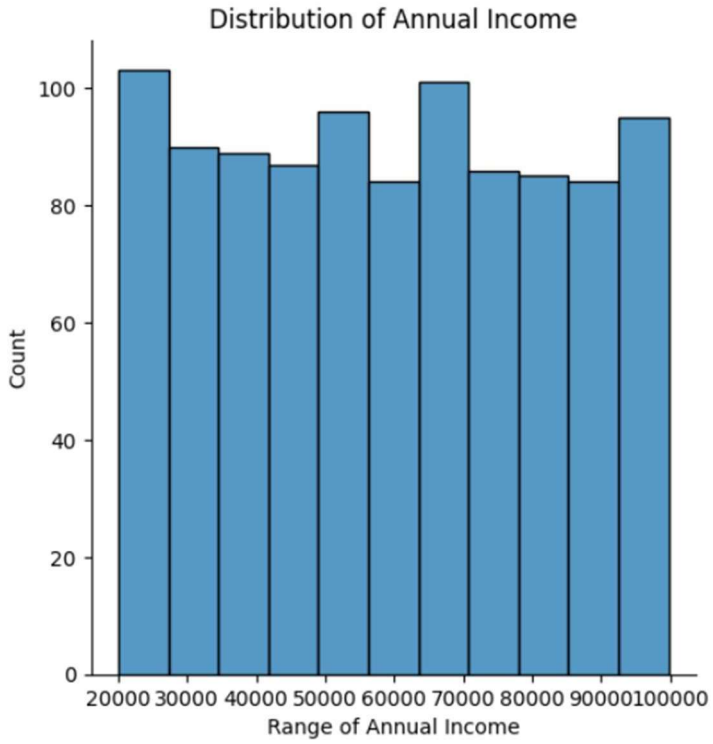
```
Text(0.5, 9.444444444444438, 'Range of Age')
```



Distribution of Annual Income

```
sns.displot(data['annual_income'])
plt.title('Distribution of Annual Income')
plt.xlabel('Range of Annual Income')
```

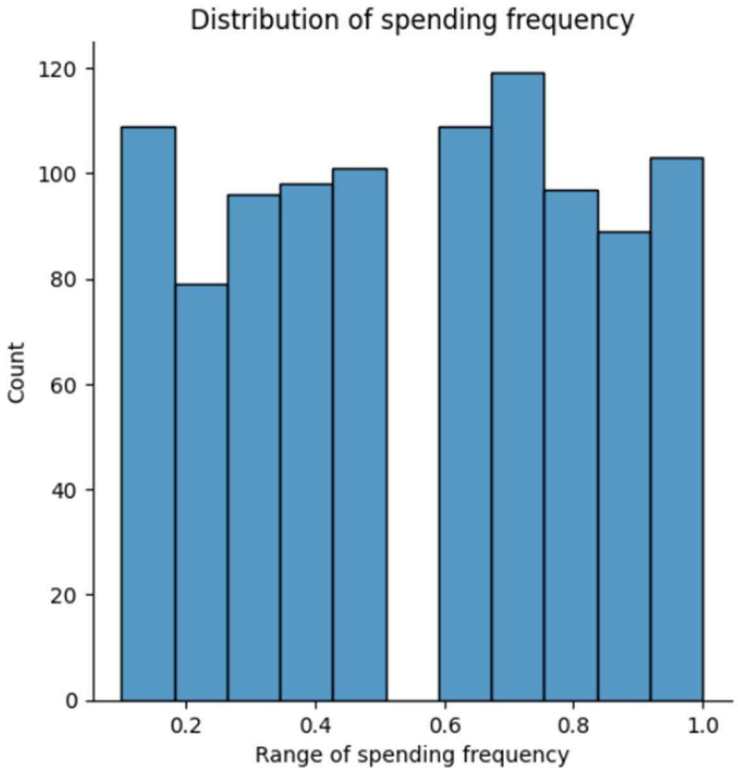
```
Text(0.5, 9.444444444444438, 'Range of Annual Income')
```



Distribution of Spending Frequency

```
sns.displot(data['spending_frequency'])
plt.title('Distribution of spending frequency')
plt.xlabel('Range of spending frequency')
```

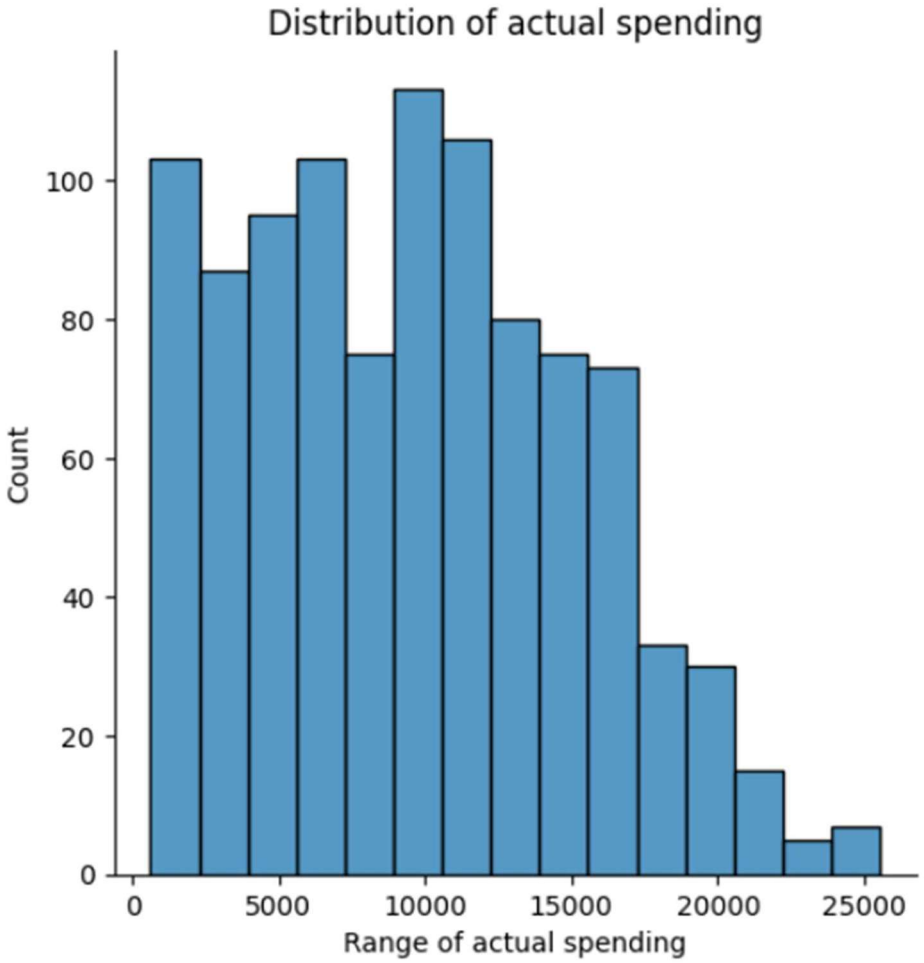
```
Text(0.5, 9.444444444444438, 'Range of spending frequency')
```



Distribution of Actual Spending

```
plt.figure(figsize=(5,3))
sns.displot(data['actual_spending'])
plt.title('Distribution of actual spending')
plt.xlabel('Range of actual spending')
```

```
Text(0.5, 9.444444444444438, 'Range of actual spending')
<Figure size 500x300 with 0 Axes>
```



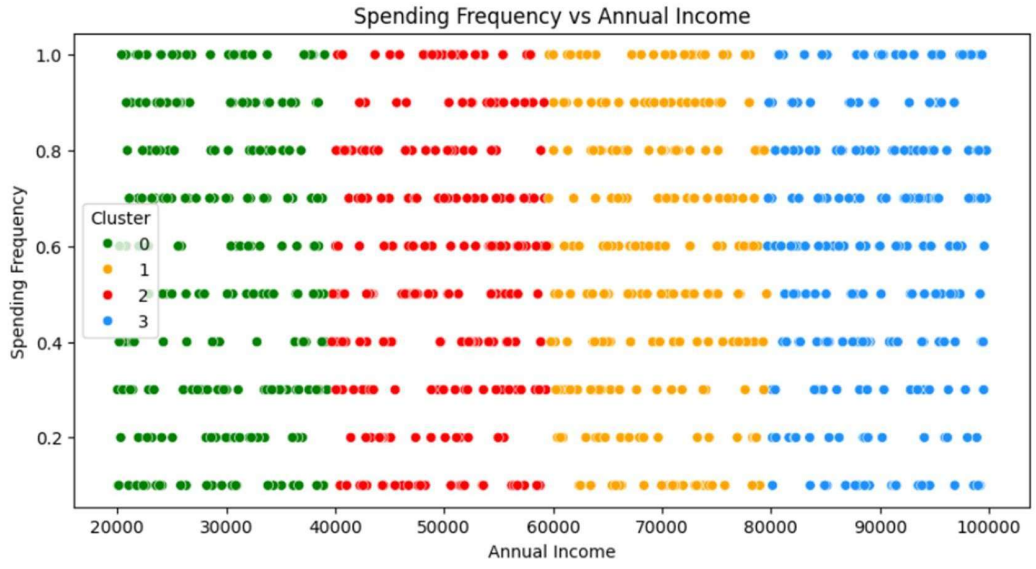
Scatterplot For Clustering

```
plt.figure(figsize=(10,5))
sns.scatterplot(x = 'annual_income', y = 'spending_frequency', data = data)
plt.xlabel('Annual Income')
plt.ylabel('Spending Frequency')
plt.title('Spending Frequency vs Annual Income')
plt.show()
```

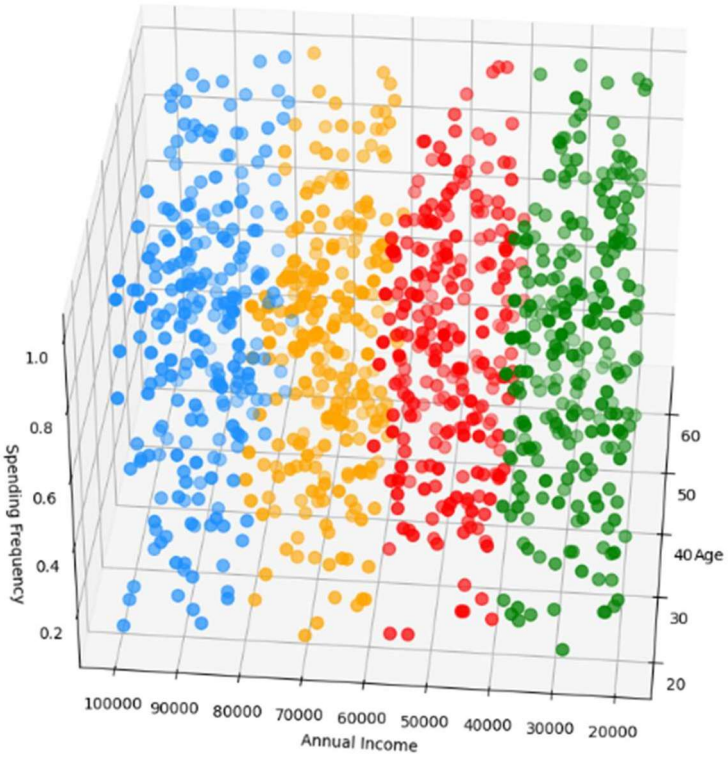
2D Model Of Clustering Using Scatterplot

2D Model

```
plt.figure(figsize=(10,5))
sns.scatterplot(x = 'annual_income',y = 'spending_frequency',
                palette=['green','orange','red','dodgerblue'],
                legend='full',data = data,hue='Cluster')
plt.xlabel('Annual Income')
plt.ylabel('Spending Frequency')
plt.title('Spending Frequency vs Annual Income')
plt.show()
```



3D Model Of Clustering Using Scatterplot



Elbow Method

Elbow Method

```
WCSS = []
for k in range(1, 11):
    kmeans = KMeans(n_clusters=k)
    kmeans.fit(scaled_df)
    WCSS.append(kmeans.inertia_)
```

```
plt.figure(figsize=(10,5))
plt.plot(range(1, 11), WCSS, marker='o')
plt.xlabel('K Value')
plt.ylabel('Within-Cluster Sum of Squares (WCSS)')
plt.title('Elbow Method for Optimal k')
plt.grid(True)
plt.show()
```