sales-marketing-campaign

November 5, 2024

[1]:
```
import pandas as pd
data = pd.read_csv('C:\\Users\\ANUSHA\\Downloads\\sales_marketing_campaign_data.
 ↪csv')
```

[22]:
```
print(data.head())
print(data.info())
print(data.describe())
```

```
   Customer_ID Campaign_ID Region  Age  Gender Income_Level  \
0            1  Campaign_A  North   60    Male          Low
1            2  Campaign_B   West   38    Male       Medium
2            3  Campaign_A   West   23  Female       Medium
3            4  Campaign_B   East   57  Female          Low
4            5  Campaign_A  South   24    Male       Medium

   Sales_Before_Campaign  Sales_After_Campaign  Engagement_Level  \
0                 913.39                778.45                 8
1                 565.49                901.61                 9
2                 125.65               1192.04                 6
3                 686.29                766.30                 3
4                 808.22                126.44                 4

   Purchase_Frequency Age_Group
0                   4     46-65
1                   2     31-45
2                   3     18-30
3                   4     46-65
4                   1     18-30
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 100 entries, 0 to 99
Data columns (total 11 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   Customer_ID            100 non-null    int64
 1   Campaign_ID            100 non-null    object
 2   Region                 100 non-null    object
 3   Age                    100 non-null    int64
 4   Gender                 100 non-null    object
```

```
 5   Income_Level           100 non-null    object
 6   Sales_Before_Campaign  100 non-null    float64
 7   Sales_After_Campaign   100 non-null    float64
 8   Engagement_Level       100 non-null    int64
 9   Purchase_Frequency     100 non-null    int64
 10  Age_Group              100 non-null    category
dtypes: category(1), float64(2), int64(4), object(4)
memory usage: 8.2+ KB
None
       Customer_ID         Age  Sales_Before_Campaign  Sales_After_Campaign  \
count   100.000000  100.000000              100.00000             100.00000
mean     50.500000   40.030000              549.01510             872.16110
std      29.011492   13.411479              252.02248             398.42552
min       1.000000   19.000000              101.15000             126.44000
25%      25.750000   28.750000              347.31500             554.58750
50%      50.500000   41.500000              540.67500             894.50500
75%      75.250000   50.250000              725.65500            1256.27500
max     100.000000   64.000000              998.92000            1493.57000

       Engagement_Level  Purchase_Frequency
count        100.000000          100.000000
mean           5.100000            3.080000
std            2.830212            1.375691
min            1.000000            1.000000
25%            2.750000            2.000000
50%            5.000000            3.000000
75%            7.000000            4.000000
max           10.000000            5.000000
```

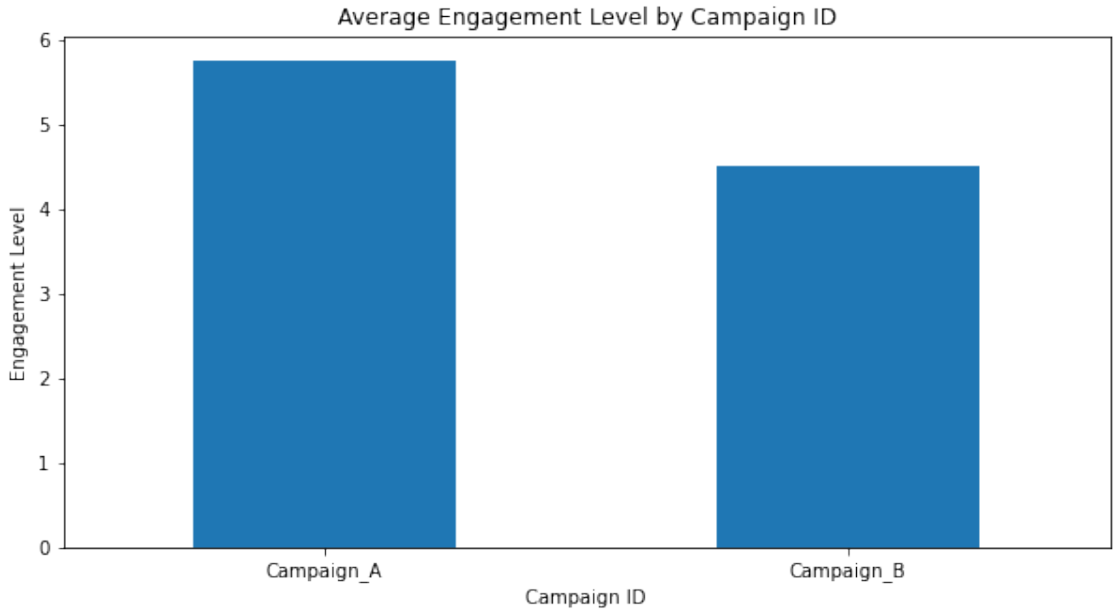
[3]:
```
print(data.isnull().sum())
```

```
Customer_ID             0
Campaign_ID             0
Region                  0
Age                     0
Gender                  0
Income_Level            0
Sales_Before_Campaign   0
Sales_After_Campaign    0
Engagement_Level        0
Purchase_Frequency      0
dtype: int64
```

[20]:
```
# Engagement level by Campaign ID
engagement_comparison = data.groupby('Campaign_ID')['Engagement_Level'].mean()
print(engagement_comparison)
engagement_comparison.plot(kind='bar', figsize=(10, 5))
```

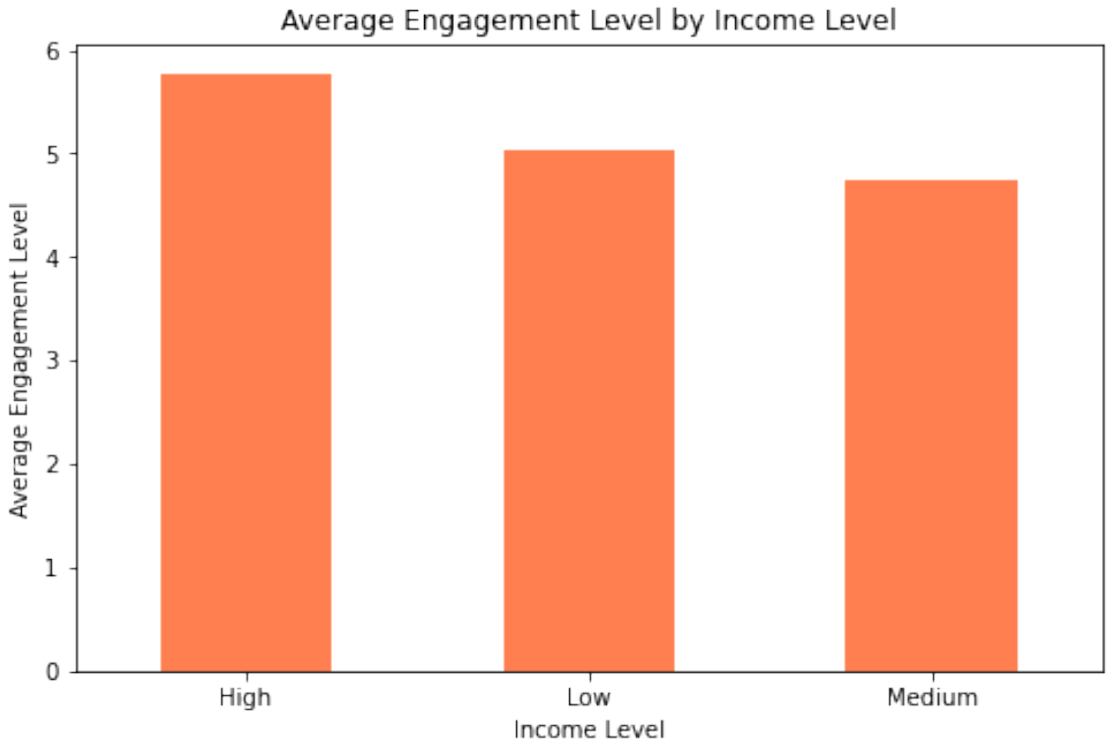
```
plt.title('Average Engagement Level by Campaign ID')
plt.ylabel('Engagement Level')
plt.xlabel('Campaign ID')
plt.xticks(rotation=0)
plt.show()
```

```
Campaign_ID
Campaign_A    5.75
Campaign_B    4.50
Name: Engagement_Level, dtype: float64
```



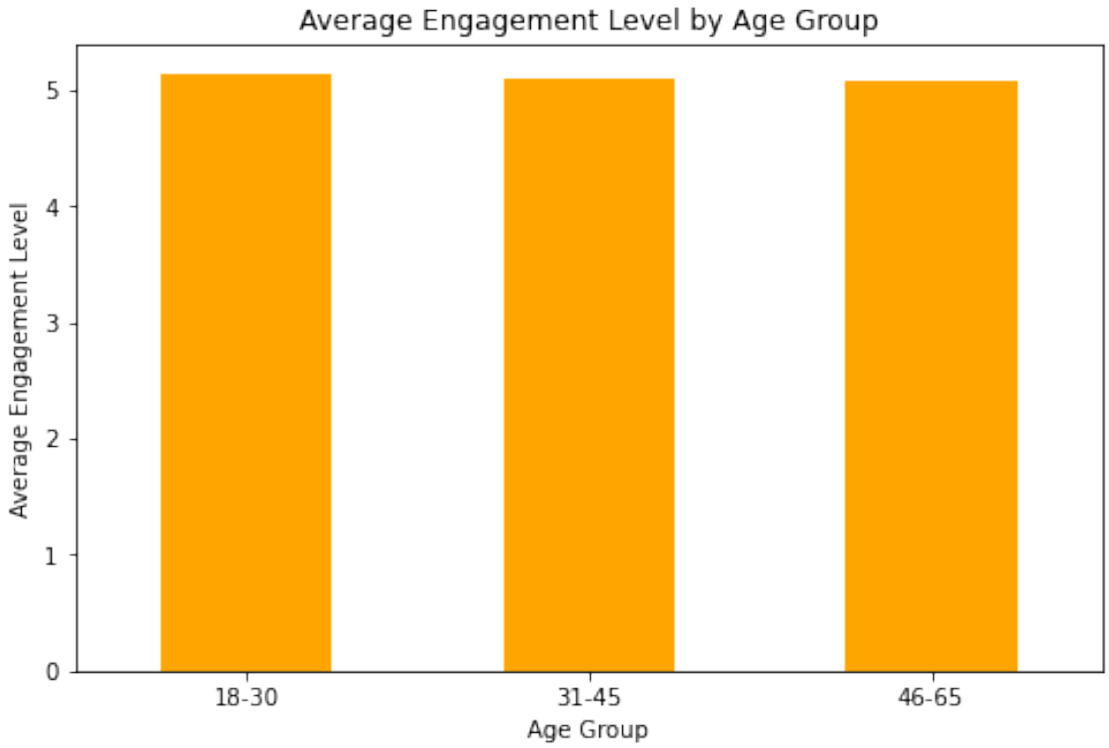
[14]:
```
# Engagement level by income level
engagement_by_income = data.groupby('Income_Level')['Engagement_Level'].mean()
print(engagement_by_income)
engagement_by_income.plot(kind='bar', color='coral', figsize=(8, 5))
plt.title('Average Engagement Level by Income Level')
plt.xlabel('Income Level')
plt.ylabel('Average Engagement Level')
plt.xticks(rotation=0)
plt.show()
```

```
Income_Level
High      5.769231
Low       5.032258
Medium    4.744186
Name: Engagement_Level, dtype: float64
```



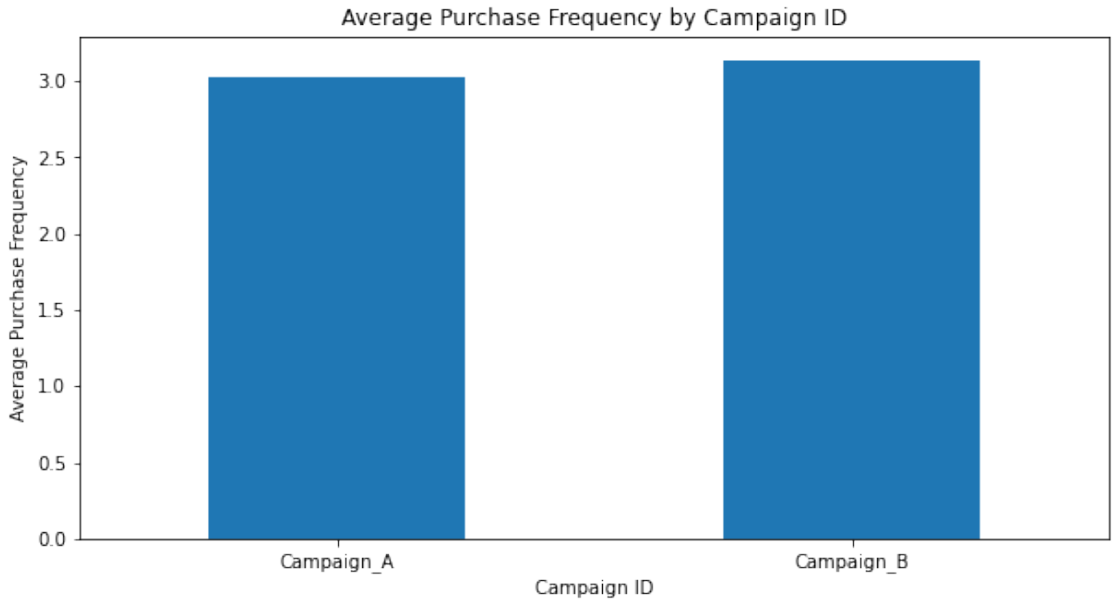
[23]:
```
bins = [18, 30, 45, 65]
labels = ['18-30', '31-45', '46-65']
data['Age_Group'] = pd.cut(data['Age'], bins=bins, labels=labels, right=False)
# Calculate mean engagement level by age group
engagement_by_age_group = data.groupby('Age_Group')['Engagement_Level'].mean()
print(engagement_by_age_group)
engagement_by_age_group.plot(kind='bar', color='orange', figsize=(8, 5))
plt.title('Average Engagement Level by Age Group')
plt.xlabel('Age Group')
plt.ylabel('Average Engagement Level')
plt.xticks(rotation=0)
plt.show()
```

```
Age_Group
18-30    5.142857
31-45    5.096774
46-65    5.073171
Name: Engagement_Level, dtype: float64
```



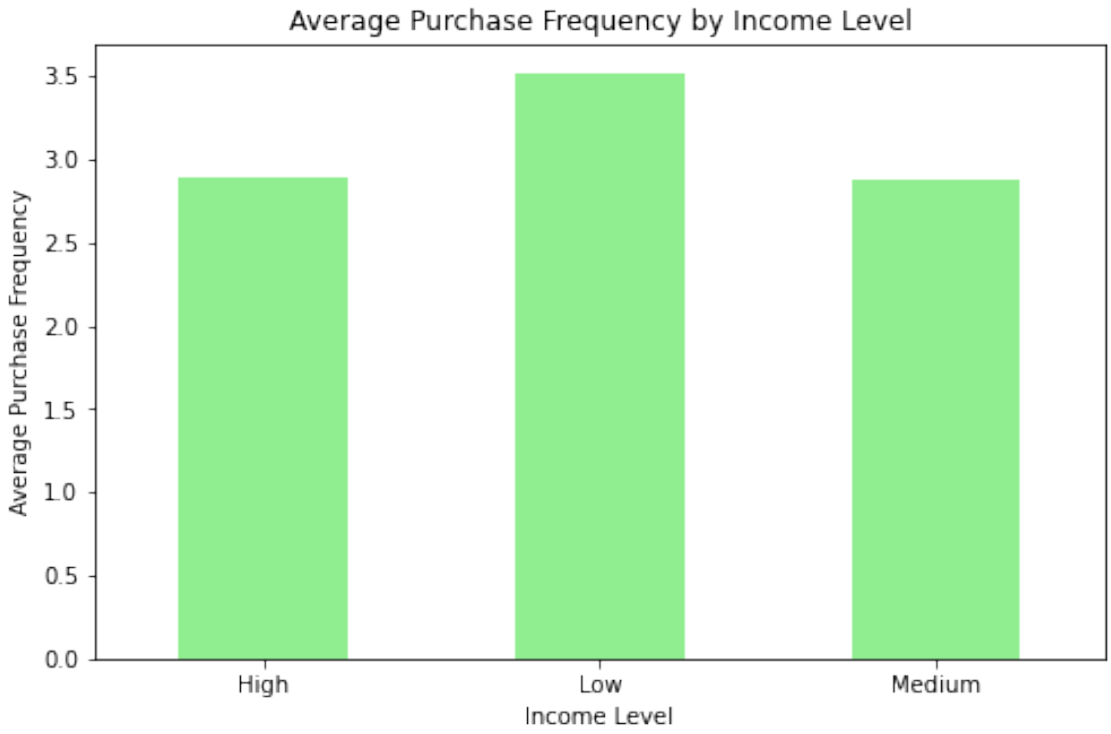
[17]:
```
# purchase frequency by Campaign ID
purchase_frequency_comparison = data.
  ↪groupby('Campaign_ID')['Purchase_Frequency'].mean()
print(purchase_frequency_comparison)
purchase_frequency_comparison.plot(kind='bar', figsize=(10, 5))
plt.title('Average Purchase Frequency by Campaign ID')
plt.ylabel('Average Purchase Frequency')
plt.xlabel('Campaign ID')
plt.xticks(rotation=0)
plt.show()
```

```
Campaign_ID
Campaign_A    3.020833
Campaign_B    3.134615
Name: Purchase_Frequency, dtype: float64
```



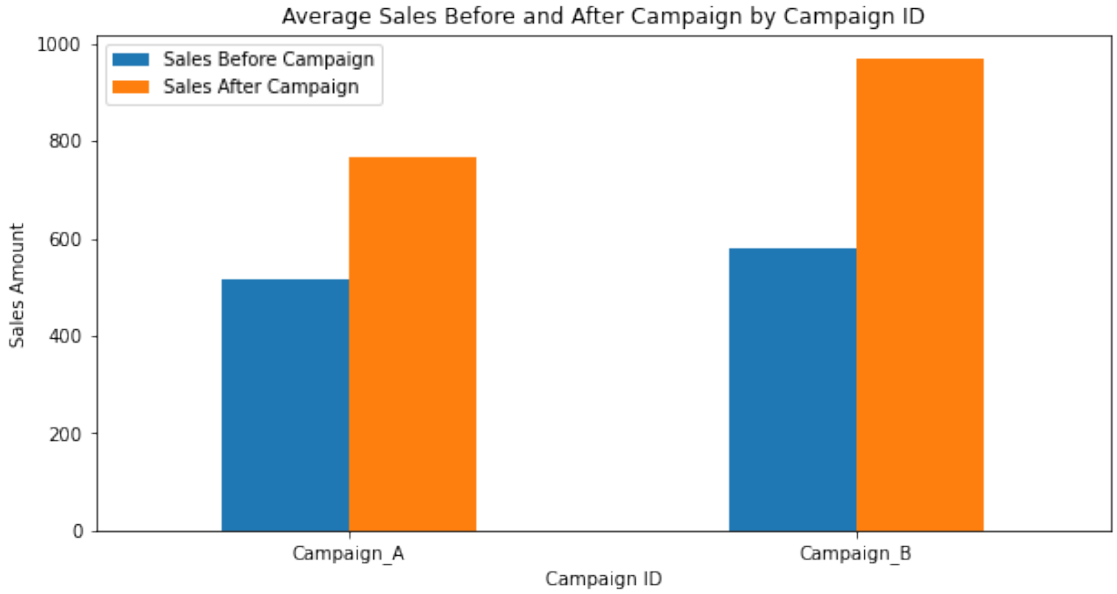
[13]:
```
# purchase frequency by income level
purchase_freq_by_income = data.groupby('Income_Level')['Purchase_Frequency'].
  ↪mean()
print(purchase_freq_by_income)
purchase_freq_by_income.plot(kind='bar', color='lightgreen', figsize=(8, 5))
plt.title('Average Purchase Frequency by Income Level')
plt.xlabel('Income Level')
plt.ylabel('Average Purchase Frequency')
plt.xticks(rotation=0)
plt.show()
```

```
Income_Level
High      2.884615
Low       3.516129
Medium    2.883721
Name: Purchase_Frequency, dtype: float64
```



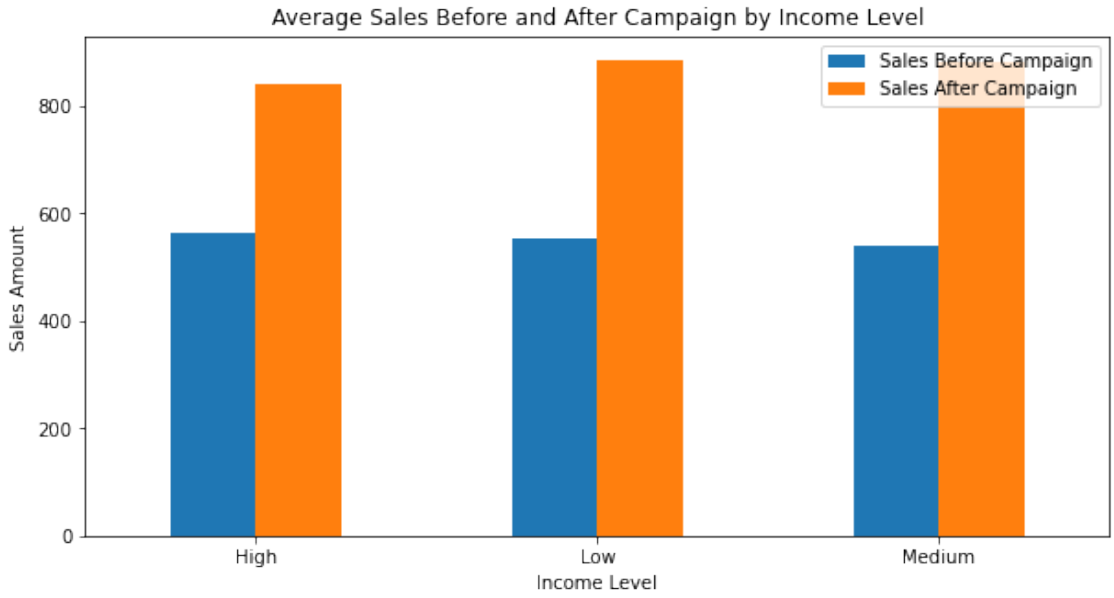
[25]:
```
sales_comparison = data.groupby('Campaign_ID')[['Sales_Before_Campaign',␣
↪'Sales_After_Campaign']].mean()
print(sales_comparison)
import matplotlib.pyplot as plt
sales_comparison.plot(kind='bar', figsize=(10, 5))
plt.title('Average Sales Before and After Campaign by Campaign ID')
plt.ylabel('Sales Amount')
plt.xlabel('Campaign ID')
plt.xticks(rotation=0)
plt.legend(['Sales Before Campaign', 'Sales After Campaign'])
plt.show()
```

```
             Sales_Before_Campaign  Sales_After_Campaign
Campaign_ID
Campaign_A              516.548750            767.348333
Campaign_B              578.984038            968.911346
```



[18]:
```
income_sales_comparison = data.
  ↪groupby('Income_Level')[['Sales_Before_Campaign', 'Sales_After_Campaign']].
  ↪mean()
print(income_sales_comparison)
income_sales_comparison.plot(kind='bar', figsize=(10, 5))
plt.title('Average Sales Before and After Campaign by Income Level')
plt.ylabel('Sales Amount')
plt.xlabel('Income Level')
plt.xticks(rotation=0)
plt.legend(['Sales Before Campaign', 'Sales After Campaign'])
plt.show()
```

```
              Sales_Before_Campaign  Sales_After_Campaign
Income_Level
High                     562.165000            842.155000
Low                      552.051613            885.094839
Medium                   538.874884            880.980000
```



[15]:
```
from scipy.stats import f_oneway

# Extract engagement levels for each income group
high_engagement = data[data['Income_Level'] == 'High']['Engagement_Level']
medium_engagement = data[data['Income_Level'] == 'Medium']['Engagement_Level']
low_engagement = data[data['Income_Level'] == 'Low']['Engagement_Level']

# ANOVA test
f_stat, p_val = f_oneway(high_engagement, medium_engagement, low_engagement)
print("ANOVA test for Engagement Level by Income Level: F-statistic =", f_stat,␣
 ↪", p-value =", p_val)

#p-value < 0.05, it suggests significant differences in engagement levels␣
 ↪across income groups.
```

```
ANOVA test for Engagement Level by Income Level: F-statistic =
1.0772471187142683 , p-value = 0.34457005721603334
```