

bike-sales-analysis

November 29, 2025

```
[9]: import pandas as pd  
import numpy as np  
import matplotlib.pyplot as plt
```

```
[14]: import pandas as pd  
  
file_path = r"D:\intern2\bike deko\Raw Data.xlsx"  
df = pd.read_excel(file_path)  
  
print(df.head())
```

	ID	Marital Status	Gender	Income	Children	Education	\
0	12496		M	F	40000	1	Bachelors
1	24107		M	M	30000	3	Partial College
2	14177		M	M	80000	5	Partial College
3	24381		S	M	70000	0	Bachelors
4	25597		S	M	30000	0	Bachelors

	Occupation	Home Owner	Cars	Commute Distance	Region	Age	\
0	Skilled Manual	Yes	0	0-1 Miles	Europe	42	
1	Clerical	Yes	1	0-1 Miles	Europe	43	
2	Professional	No	2	2-5 Miles	Europe	60	
3	Professional	Yes	1	5-10 Miles	Pacific	41	
4	Clerical	No	0	0-1 Miles	Europe	36	

	Purchased Bike
0	No
1	No
2	No
3	Yes
4	Yes

```
[15]: df = df.drop_duplicates()
```

```
[16]: df = df.fillna({  
    "Gender": "Unknown",  
    "Income": df["Income"].median()  
})
```

```
[18]: df["Income"] = df["Income"].astype(int)
df["Age"] = df["Age"].astype(int)
```

```
[19]: df["Age_Group"] = pd.cut(
    df["Age"],
    bins=[0, 25, 35, 45, 60, 100],
    labels=["<25", "25-35", "35-45", "45-60", "60+"]
)
```

```
[20]: df["Gender"] = df["Gender"].fillna("Unknown")
df["Income"] = df["Income"].fillna(df["Income"].median())
```

```
[21]: sales_by_gender = df[df["Purchased Bike"]=="Yes"].groupby("Gender")["ID"].
    ↪count()
print(sales_by_gender)
```

```
Gender
F      239
M      242
Name: ID, dtype: int64
```

```
[22]: region_sales = df.groupby("Region")["Purchased Bike"].
    ↪value_counts(normalize=True)
print(region_sales)
```

```
Region          Purchased Bike
Europe          No           0.506667
                  Yes          0.493333
North America  No           0.566929
                  Yes          0.433071
Pacific         Yes          0.588542
                  No           0.411458
Name: proportion, dtype: float64
```

```
[24]: print("Shape:", df.shape)
print("\nColumns:\n", df.columns)

print("\nSummary Statistics:\n")
print(df.describe())

print("\nMissing Values:\n")
print(df.isnull().sum())
```

```
Shape: (1000, 14)
```

```
Columns:
Index(['ID', 'Marital Status', 'Gender', 'Income', 'Children', 'Education',
       'Occupation', 'Home Owner', 'Cars', 'Commute Distance', 'Region', 'Age',
```

```
'Purchased Bike', 'Age_Group'],
dtype='object')
```

Summary Statistics:

	ID	Income	Children	Cars	Age
count	1000.000000	1000.000000	1000.000000	1000.000000	1000.000000
mean	19965.992000	56360.000000	1.898000	1.442000	44.163000
std	5347.333948	31085.635215	1.628572	1.125123	11.364488
min	11000.000000	10000.000000	0.000000	0.000000	25.000000
25%	15290.750000	30000.000000	0.000000	1.000000	35.000000
50%	19744.000000	60000.000000	2.000000	1.000000	43.000000
75%	24470.750000	70000.000000	3.000000	2.000000	52.000000
max	29447.000000	170000.000000	5.000000	4.000000	89.000000

Missing Values:

```
ID          0
Marital Status  0
Gender        0
Income         0
Children       0
Education      0
Occupation     0
Home Owner    0
Cars           0
Commute Distance 0
Region         0
Age            0
Purchased Bike 0
Age_Group      0
dtype: int64
```

```
[25]: print(df["Gender"].value_counts())
```

```
Gender
M      511
F      489
Name: count, dtype: int64
```

```
[26]: print(df["Marital Status"].value_counts())
```

```
Marital Status
M      538
S      462
Name: count, dtype: int64
```

```
[27]: print(df["Purchased Bike"].value_counts())
```

```
Purchased Bike  
No      519  
Yes     481  
Name: count, dtype: int64
```

```
[28]: bike_by_gender = df.groupby("Gender")["Purchased Bike"].value_counts()  
print(bike_by_gender)
```

```
Gender Purchased Bike  
F      No          250  
       Yes         239  
M      No          269  
       Yes         242  
Name: count, dtype: int64
```

```
[29]: bike_by_region = df.groupby("Region")["Purchased Bike"].value_counts()  
print(bike_by_region)
```

```
Region      Purchased Bike  
Europe      No          152  
           Yes         148  
North America No          288  
           Yes         220  
Pacific     Yes          113  
           No          79  
Name: count, dtype: int64
```

```
[30]: income_purchase = df.groupby("Purchased Bike")["Income"].mean()  
print(income_purchase)
```

```
Purchased Bike  
No      54874.759152  
Yes     57962.577963  
Name: Income, dtype: float64
```

```
[31]: age_purchase = df.groupby("Purchased Bike")["Age"].mean()  
print(age_purchase)
```

```
Purchased Bike  
No      45.327553  
Yes     42.906445  
Name: Age, dtype: float64
```

```
[33]: children_purchase = df.groupby("Children")["Purchased Bike"].value_counts()  
print(children_purchase)
```

```
Children Purchased Bike  
0        Yes          142
```

```
No          139  
1  Yes        97  
    No         72  
2  No          112  
    Yes        97  
3  Yes         73  
    No         61  
4  No          72  
    Yes        54  
5  No          63  
    Yes        18
```

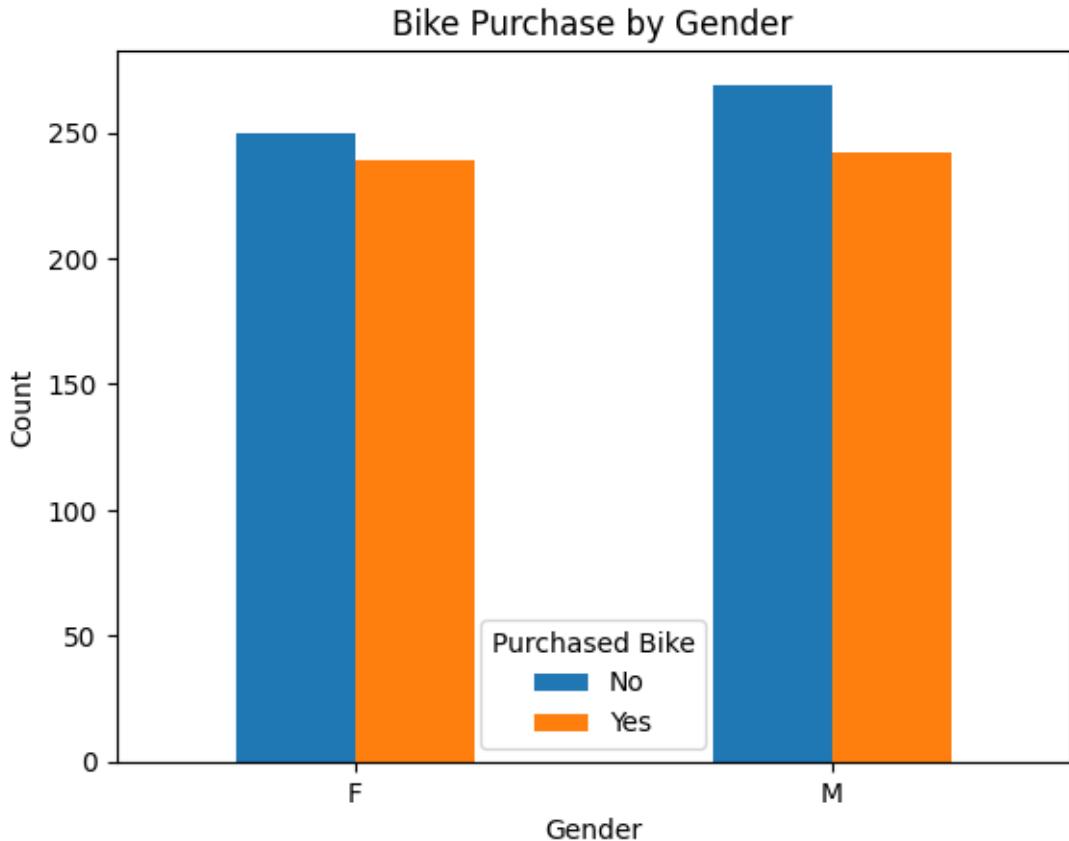
Name: count, dtype: int64

```
[35]: age_group_purchase = df.groupby("Age_Group", observed=False)[ "Purchased Bike"] .  
      ↪value_counts()  
print(age_group_purchase)
```

Age_Group	Purchased Bike	
<25	Yes	4
	No	2
25–35	No	139
	Yes	111
35–45	Yes	188
	No	138
45–60	No	170
	Yes	148
60+	No	70
	Yes	30

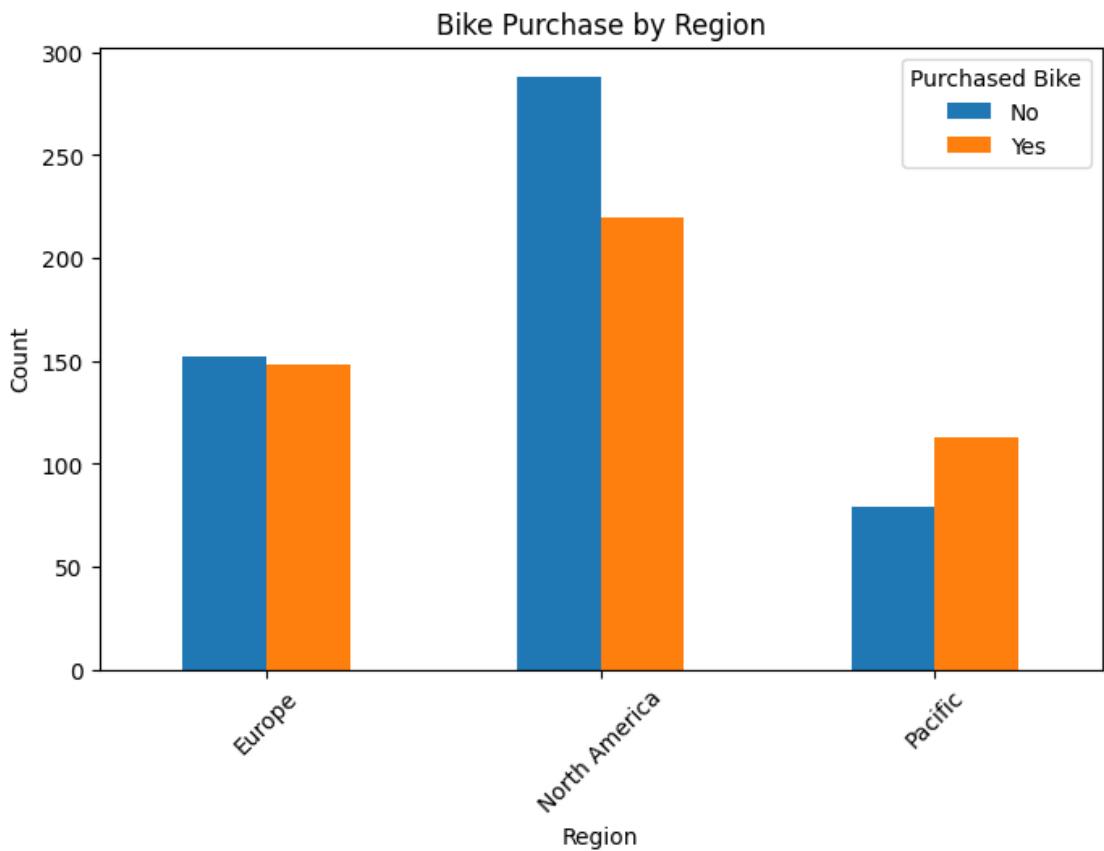
Name: count, dtype: int64

```
[36]: import matplotlib.pyplot as plt  
  
purchase_gender = df.groupby("Gender") [ "Purchased Bike"] .value_counts() .  
      ↪unstack()  
  
purchase_gender.plot(kind="bar")  
plt.title("Bike Purchase by Gender")  
plt.xlabel("Gender")  
plt.ylabel("Count")  
plt.xticks(rotation=0)  
plt.show()
```



```
[37]: region_purchase = df.groupby("Region")["Purchased Bike"].value_counts().unstack()

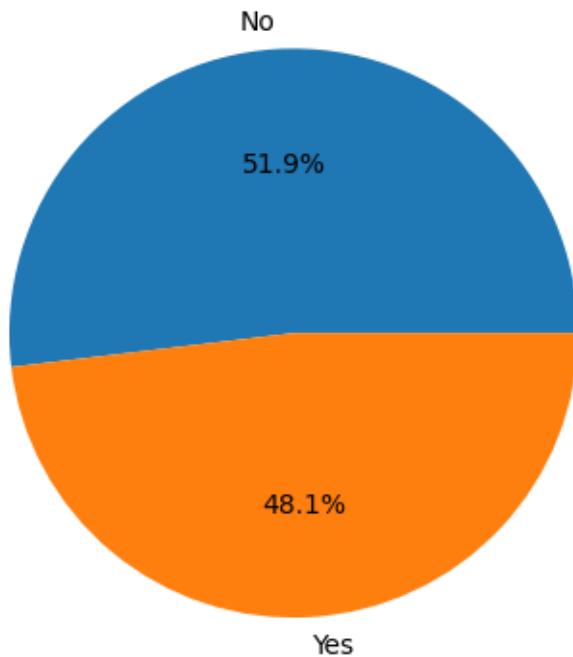
region_purchase.plot(kind="bar", figsize=(8,5))
plt.title("Bike Purchase by Region")
plt.xlabel("Region")
plt.ylabel("Count")
plt.xticks(rotation=45)
plt.show()
```



```
[38]: purchase_counts = df["Purchased Bike"].value_counts()

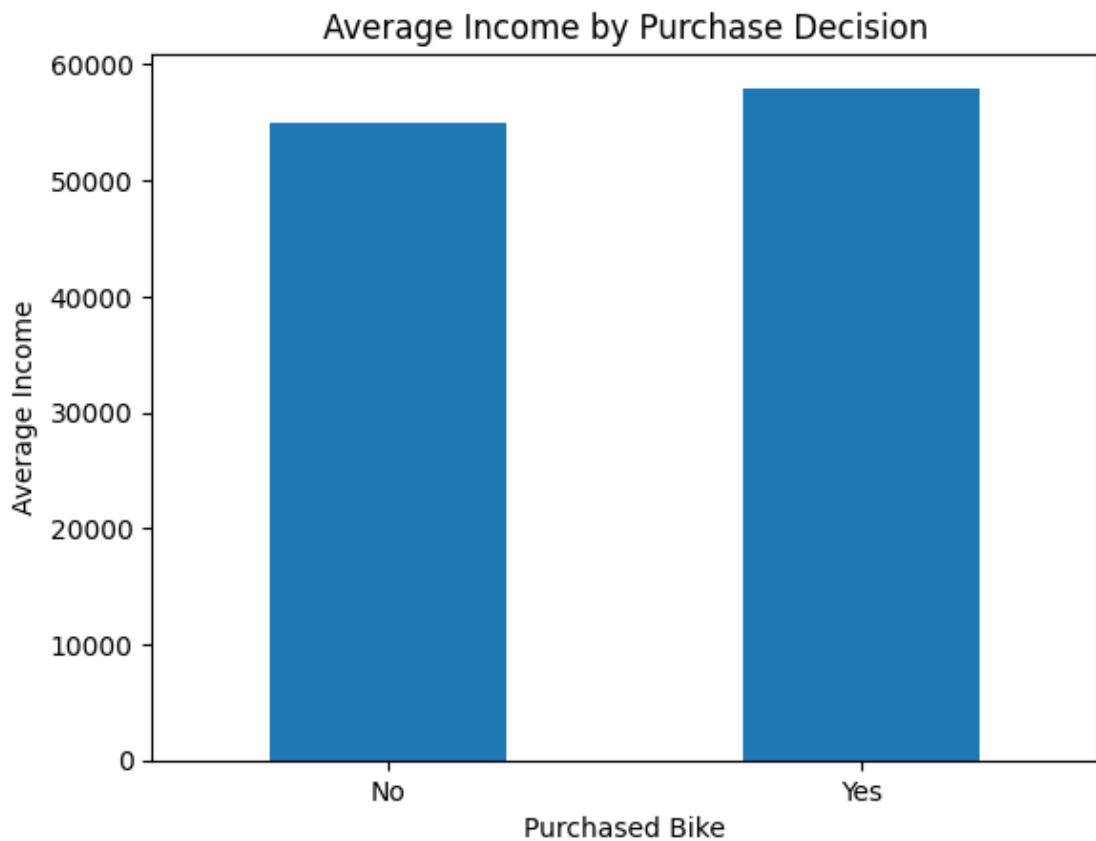
plt.pie(purchase_counts, labels=purchase_counts.index, autopct='%1.1f%%')
plt.title("Overall Purchase Distribution")
plt.show()
```

Overall Purchase Distribution

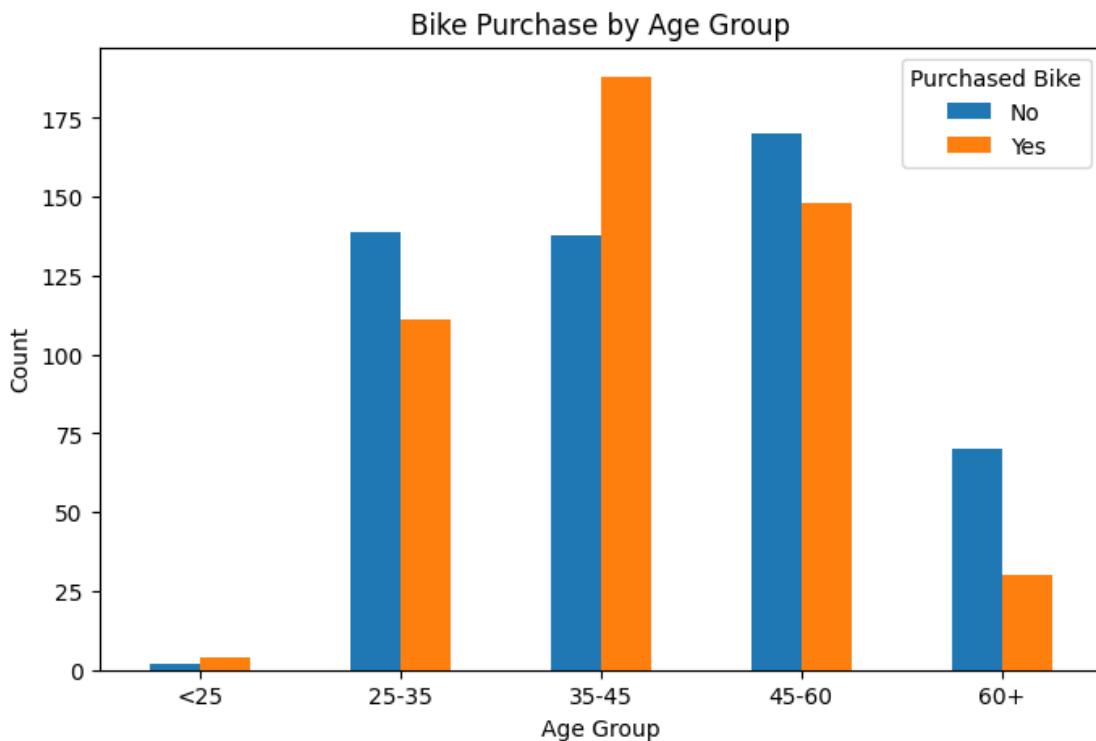


```
[39]: income_purchase = df.groupby("Purchased Bike")["Income"].mean()

income_purchase.plot(kind="bar")
plt.title("Average Income by Purchase Decision")
plt.xlabel("Purchased Bike")
plt.ylabel("Average Income")
plt.xticks(rotation=0)
plt.show()
```



```
[41]: age_group_purchase = df.groupby(["Age_Group"], observed=True)[["Purchased Bike"]].  
      ↵value_counts().unstack()  
  
age_group_purchase.plot(kind="bar", figsize=(8,5))  
plt.title("Bike Purchase by Age Group")  
plt.xlabel("Age Group")  
plt.ylabel("Count")  
plt.xticks(rotation=0)  
plt.show()
```



```
[42]: output_path = r"D:\intern2\bike deko\processed data of bike deko.xlsx"

df.to_excel(output_path, index=False)

print("File saved successfully at:", output_path)
```

File saved successfully at: D:\intern2\bike deko\processed data of bike deko.xlsx

```
[ ]:
```