# Polars Descriptive Statistics Script

## Data Loading

In [ ]:
```
import polars as pl

marketing_data = pl.read_csv("mydata/ifood_df.csv")
```

## Data Manipulation

In [ ]:
```
marketing_data = marketing_data.with_columns(
    TotalSpending =
    pl.col("MntWines")
    + pl.col("MntFruits")
    + pl.col("MntMeatProducts")
    + pl.col("MntFishProducts")
    + pl.col("MntSweetProducts")
)
marketing_data = marketing_data.with_columns(
    AcceptedOffer =
    pl.col("AcceptedCmp1")
    + pl.col("AcceptedCmp2")
    + pl.col("AcceptedCmp3")
    + pl.col("AcceptedCmp4")
    + pl.col("AcceptedCmp5")
)
marketing_data = marketing_data.select([
    "Income",
    "TotalSpending",
    "AcceptedOffer",
    "MntWines",
    "MntFruits",
    "MntMeatProducts",
    "MntFishProducts",
    "MntSweetProducts",
])
```

## Data Summary

In [ ]:
```
marketing_data.describe()
```

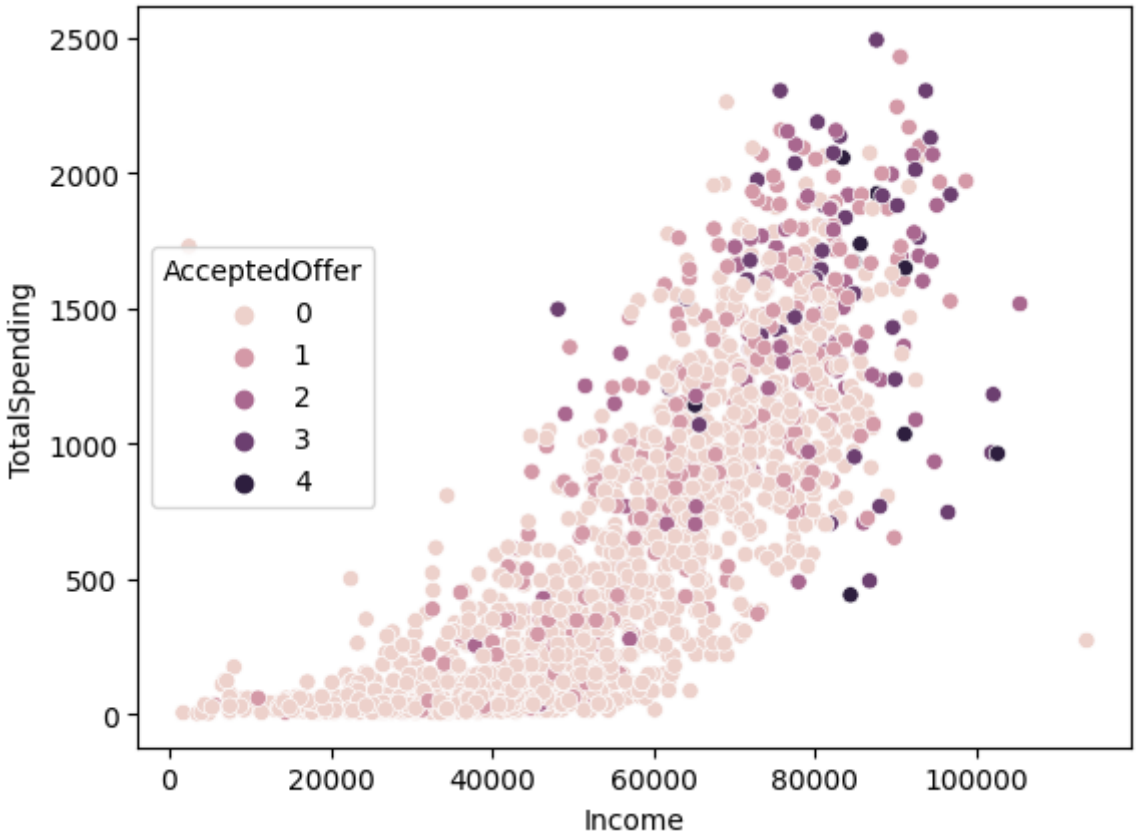
Out[ ]: shape: (9, 9)

| describe | Income | TotalSpending | AcceptedOffer | MntWines | MntFruits | MntMeatProducts | MntFishProducts | MntSweetProc |
|---|---|---|---|---|---|---|---|---|
| str | f64 | f64 | f64 | f64 | f64 | f64 | f64 | |
| "count" | 2205.0 | 2205.0 | 2205.0 | 2205.0 | 2205.0 | 2205.0 | 2205.0 | 2 |
| "null_count" | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | |
| "mean" | 51622.094785 | 562.764626 | 0.29932 | 306.164626 | 26.403175 | 165.312018 | 37.756463 | 27.12 |
| "std" | 20713.063826 | 575.936911 | 0.68044 | 337.493839 | 39.784484 | 217.784507 | 54.824635 | 41.13 |
| "min" | 1730.0 | 4.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | |
| "25%" | 35196.0 | 56.0 | 0.0 | 24.0 | 2.0 | 16.0 | 3.0 | |
| "50%" | 51287.0 | 343.0 | 0.0 | 178.0 | 8.0 | 68.0 | 12.0 | |
| "75%" | 68281.0 | 964.0 | 0.0 | 507.0 | 33.0 | 232.0 | 50.0 | |
| "max" | 113734.0 | 2491.0 | 4.0 | 1493.0 | 199.0 | 1725.0 | 259.0 | |

## Data Visualization

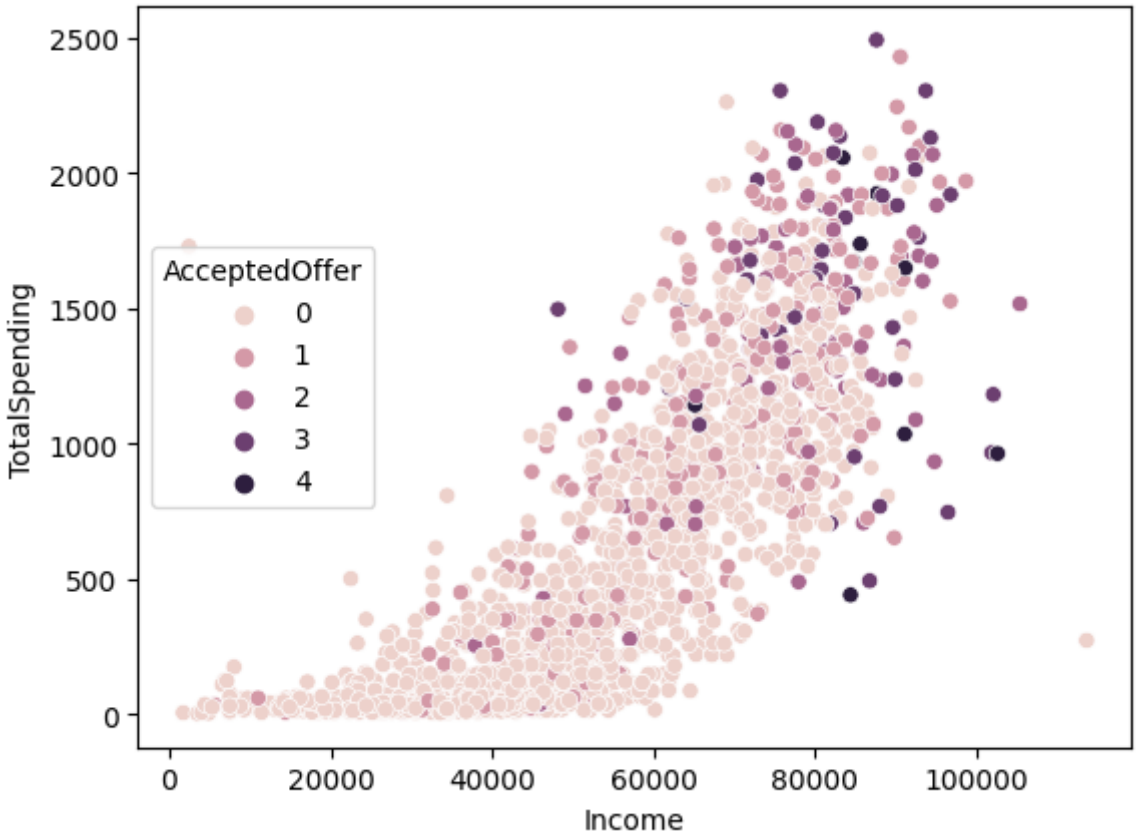
In [ ]:
```
import seaborn as sns
sns.scatterplot(data=marketing_data, x="Income", y="TotalSpending", hue="AcceptedOffer")
```

Out[ ]:
```
<Axes: xlabel='Income', ylabel='TotalSpending'>
```



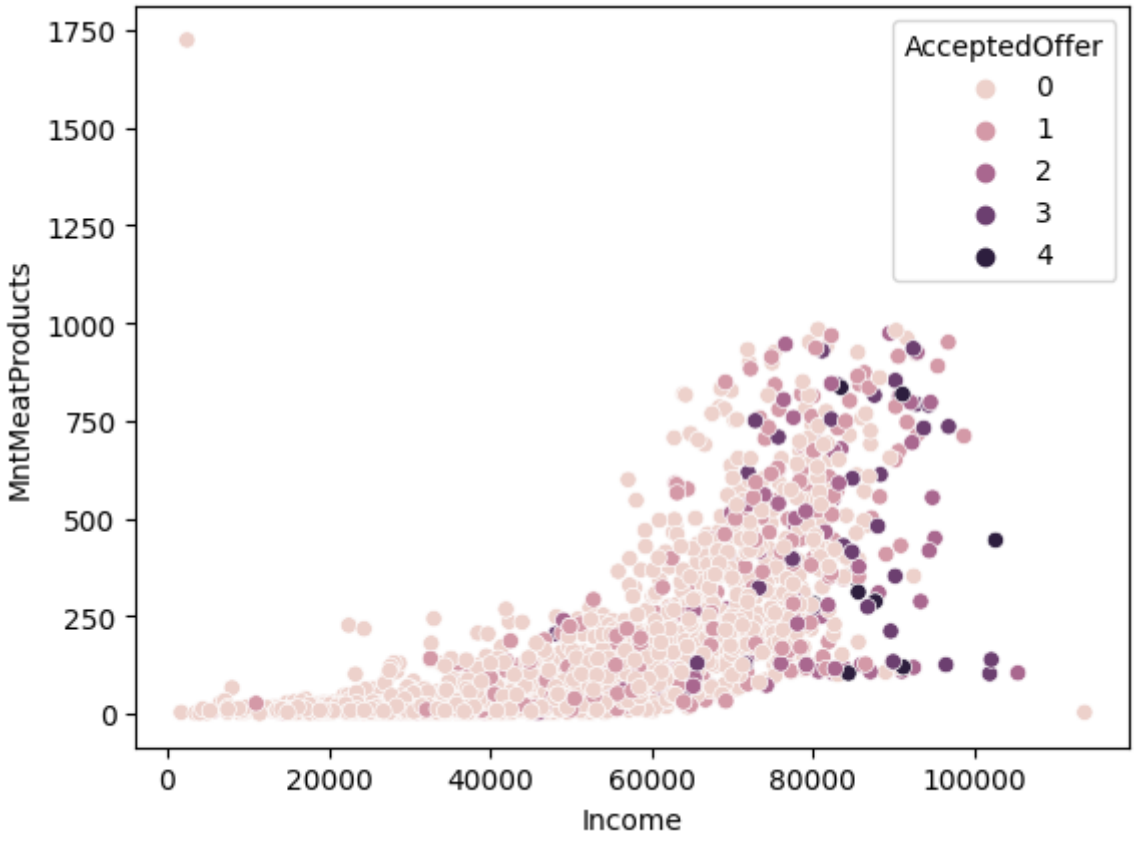

In [ ]:
```
sns.scatterplot(data=marketing_data, x="Income", y="TotalSpending", hue="AcceptedOffer")
```

Out[ ]:
```
<Axes: xlabel='Income', ylabel='TotalSpending'>
```




In [ ]:
```
sns.scatterplot(data=marketing_data, x="Income", y="MntMeatProducts", hue="AcceptedOffer")
```

Out[ ]:
```
<Axes: xlabel='Income', ylabel='MntMeatProducts'>
```


AcceptedOffer
0
1
2
3
4
MntMeatProducts
1750
1500
1250
1000
750
500
250
0
0
20000
40000
60000
80000
100000
Income