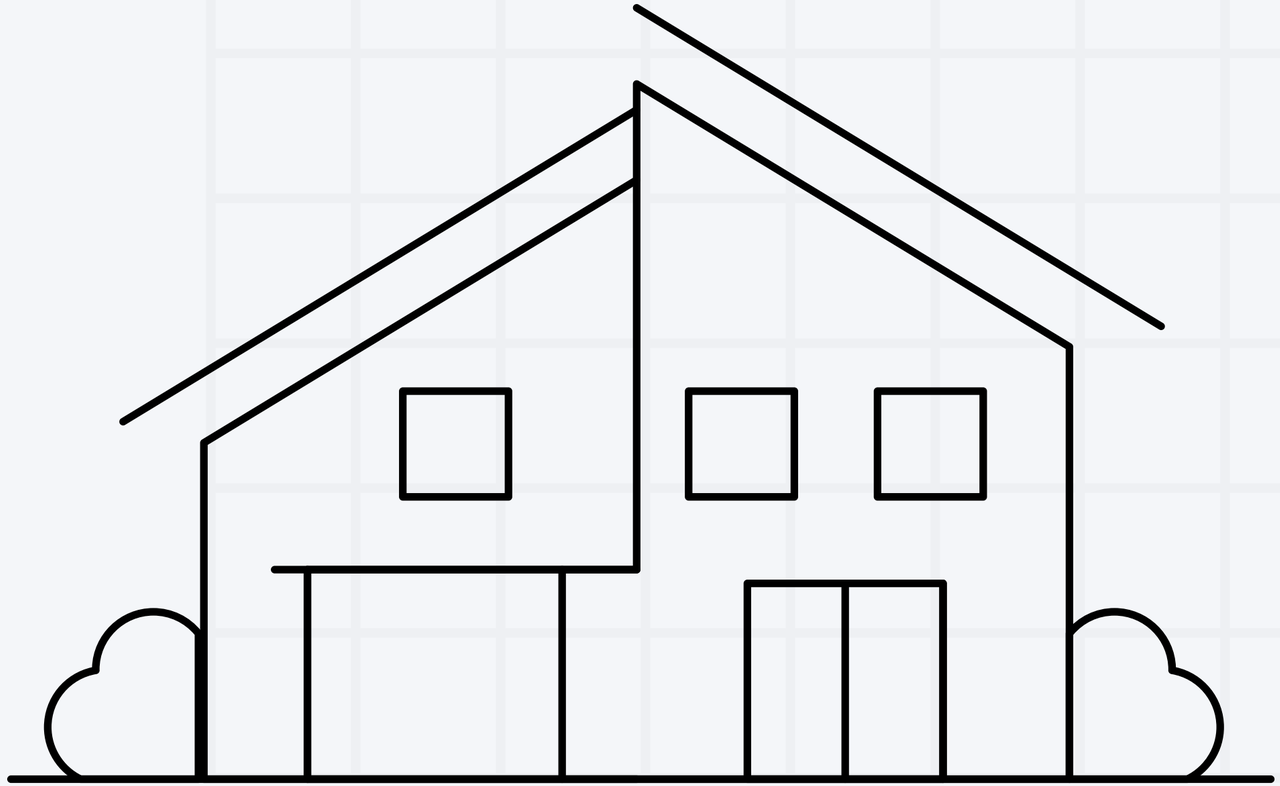
HW3

2006 micro-data survey about housing for the state of Idaho



Dhanabordee Mekintharanggur
6238077121

Q1: How many properties are worth $1,000,000 or more?

Understanding the Data

Based on the data dictionary, it is found that the property value is determined by the VAL attribute.

The VAL attribute has values including ranging from 1 to 24 and can be [EMPTY]

An empty value means that the property is not for sale and should not be considered.

Properties marked by a value of 1 have the lowest value, at less than $10000.

Properties marked by 24 has higher value than $1,000,000

```
VAL         2
    Property value
            bb .N/A (GQ/rental unit/vacant, not for sale only)
            01 .Less than $ 10000
            02 .$ 10000 - $ 14999
            03 .$ 15000 - $ 19999
            04 .$ 20000 - $ 24999
            05 .$ 25000 - $ 29999
            06 .$ 30000 - $ 34999
            07 .$ 35000 - $ 39999
            08 .$ 40000 - $ 49999
            09 .$ 50000 - $ 59999
            10 .$ 60000 - $ 69999
            11 .$ 70000 - $ 79999
            12 .$ 80000 - $ 89999
            13 .$ 90000 - $ 99999
            14 .$100000 - $124999
            15 .$125000 - $149999
            16 .$150000 - $174999
            17 .$175000 - $199999
            18 .$200000 - $249999
            19 .$250000 - $299999
            20 .$300000 - $399999
            21 .$400000 - $499999
            22 .$500000 - $749999
            23 .$750000 - $999999
            24 .$1000000+
```

Q1: How many properties are worth $1,000,000 or more?

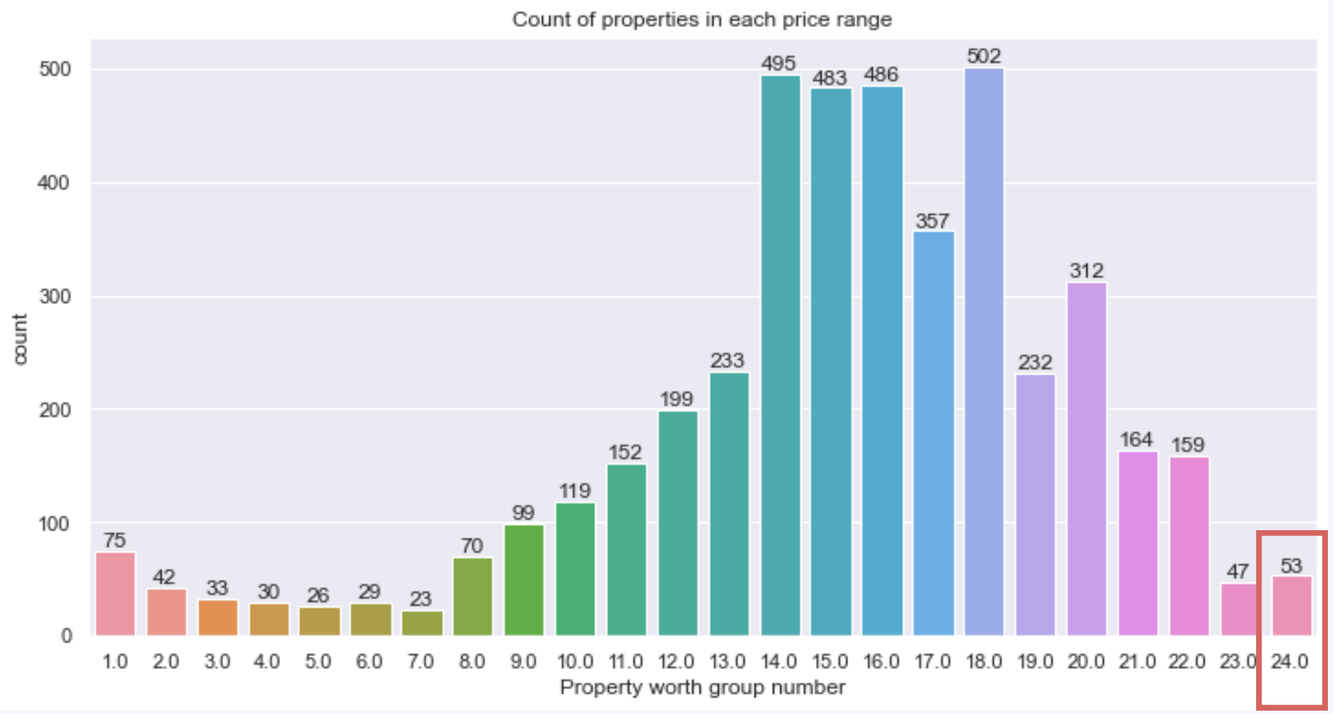
Perform check whether there are any empty values in the VAL column

```
df["VAL"].isnull().any()
```

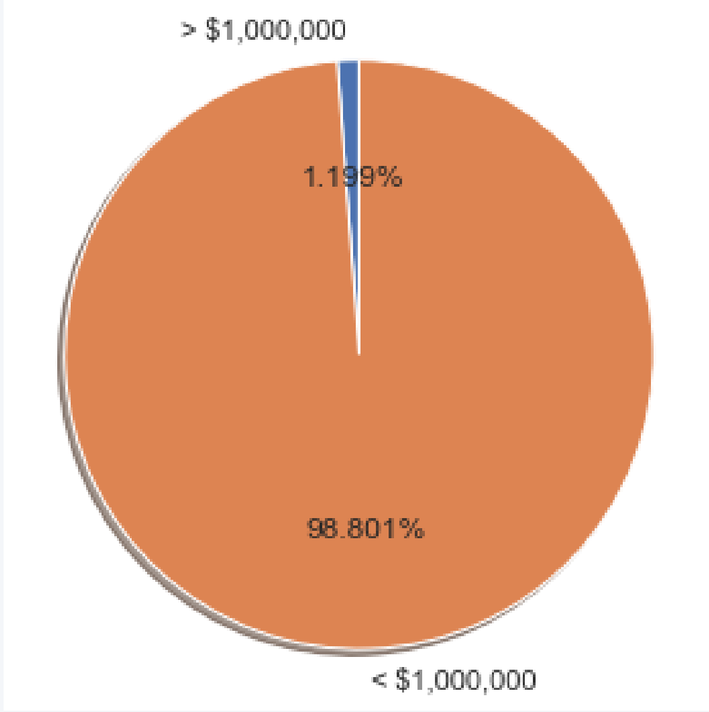
There are infact empty values; however, when using the countplot, the empty values will not be used as the X axis anyway. Thus, it is safe to continue without modifying the data.

Q1: How many properties are worth $1,000,000 or more?

By reading the dataset as CSV and calculate the size of the list of properties marked by VAL = 24, it can be found that there are a total of **53 properties that are worth more than $1,000,000**. Using Seaborn countplot, the properties' worth grouped by categories can be visualized in comparison



The pie chart further reveals that these 53 properties worthing more than $1,000,000 make up **1.2% of all properties on sale.**

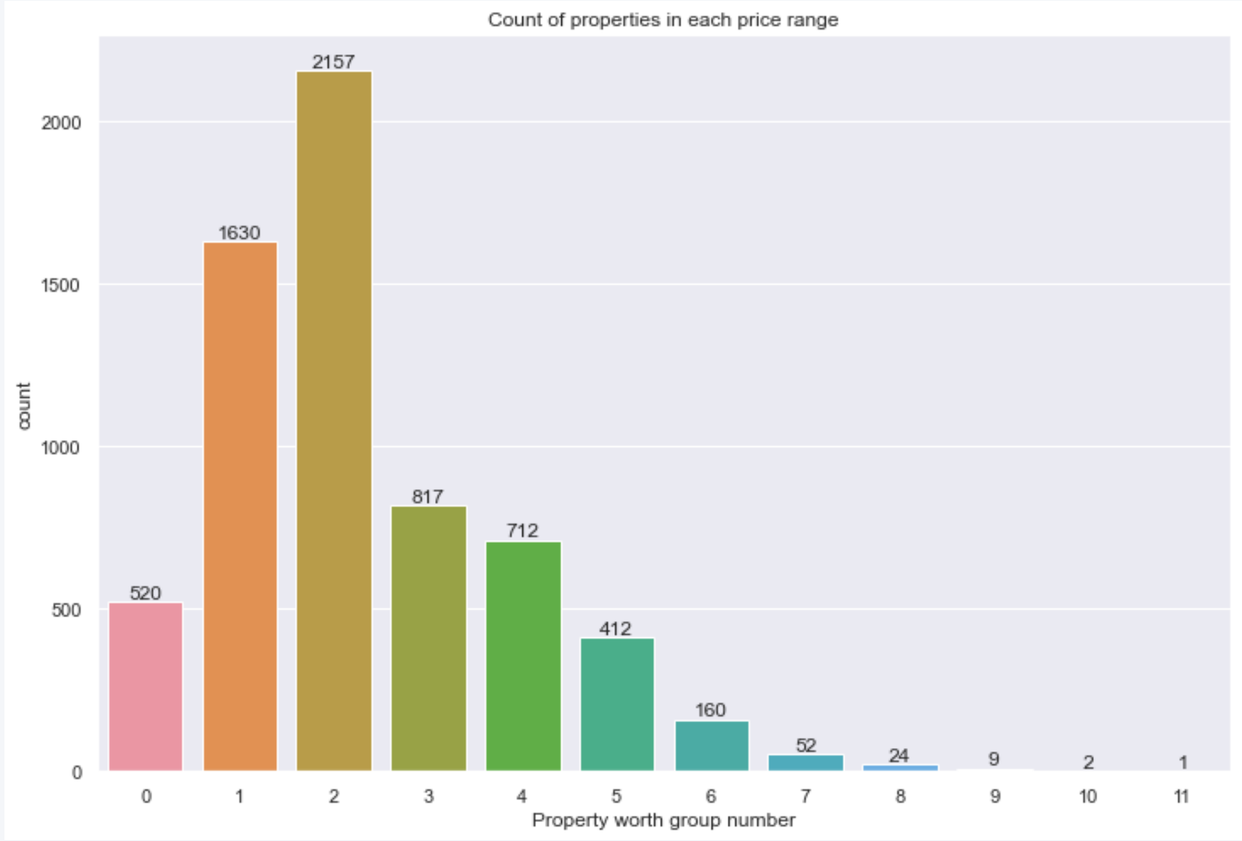


Q2: How many people recorded in a house on average?

The number of people recorded in a house is determined by the NP column according to the data dictionary.
The number directly tells the recorded number of people in each house.
This number can be used directly to calculate the average.

```
NP          2
    Number of person records following this housing record
                00 .Vacant unit
                01 .One person record (one person in household or
                   .any person in group quarters)
             02..20 .Number of person records (number of persons in
                    .household)
```

The histogram presents the distribution of the number of recorded people in a house. It can be observed that it is most common for houses to have 2 recorded people living in



Perform a check whether there are any empty values in the VAL column. Fortunately, there are no null values in the NP column

```
df["NP"].isnull().any()
✓ 0.2s
False
```

Q2: How many people recorded in a house on average?

```
number_of_person_recorded_list = df["NP"]
avg_number_of_person_recorded = np.average(number_of_person_recorded_list)
print(avg_number_of_person_recorded)
```

Understanding the data

The data is read and turned into a Pandas data frame.
Using NumPy average function, the average value of the NP column of the data frame can be calculated to be 2.2984913793103448

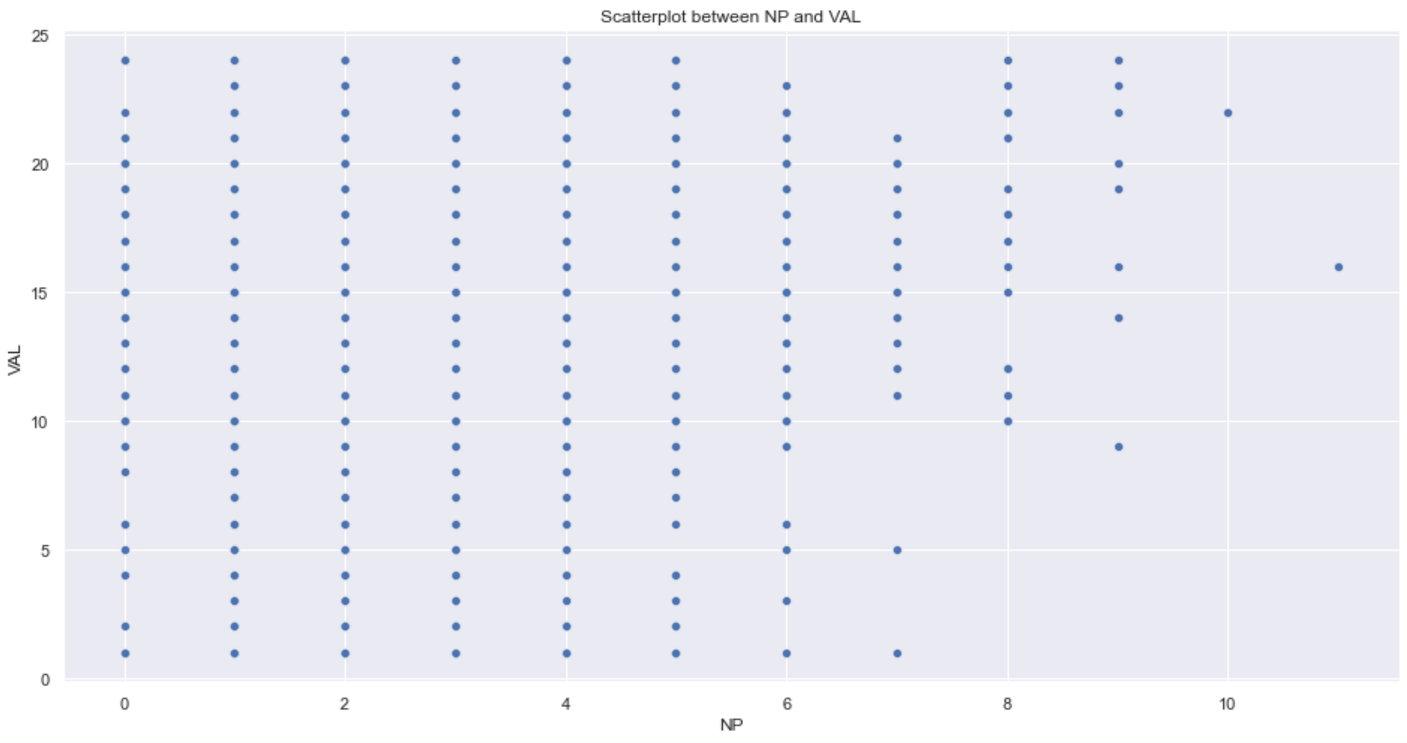
Thus, there are **on average, 2.298 people per house in Idaho.**

Q3 Draw a graph to show the relationship between the property value and the number of persons recorded

Scatterplot

The scatterplot is simple and easy to understand when compared to sets of data variables to identify their relationship.

However, from the plot, it appears to have **little to no relationship between the NP (recorded number of persons per house) and VAL(the property value)** as the plot covers most possible areas of the plot and does not show significant relationship between them.

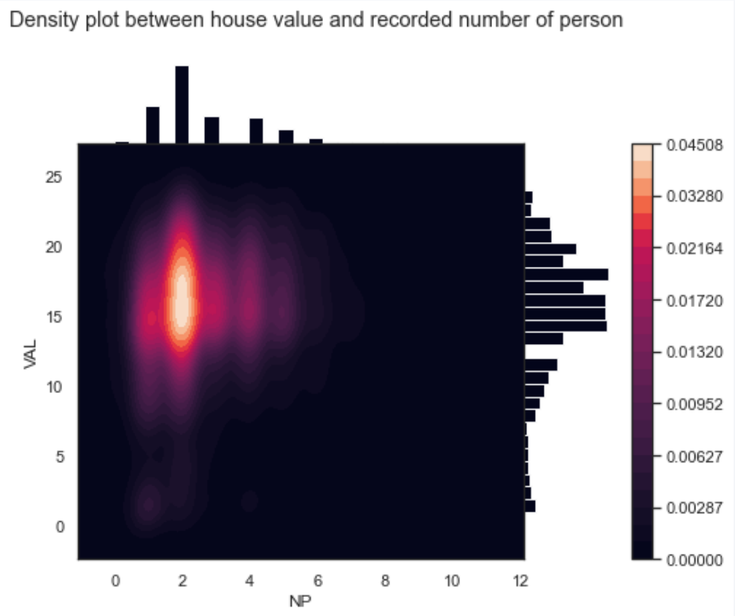


Correlation

Upon calculating the **correlation** between NP and VAL to be **0.1306154145118014 5**

This suggests that there is little to no correlation between the two columns alone.

Q3 Draw a graph to show the relationship between the property value and the number of persons recorded



Scatterplot with count

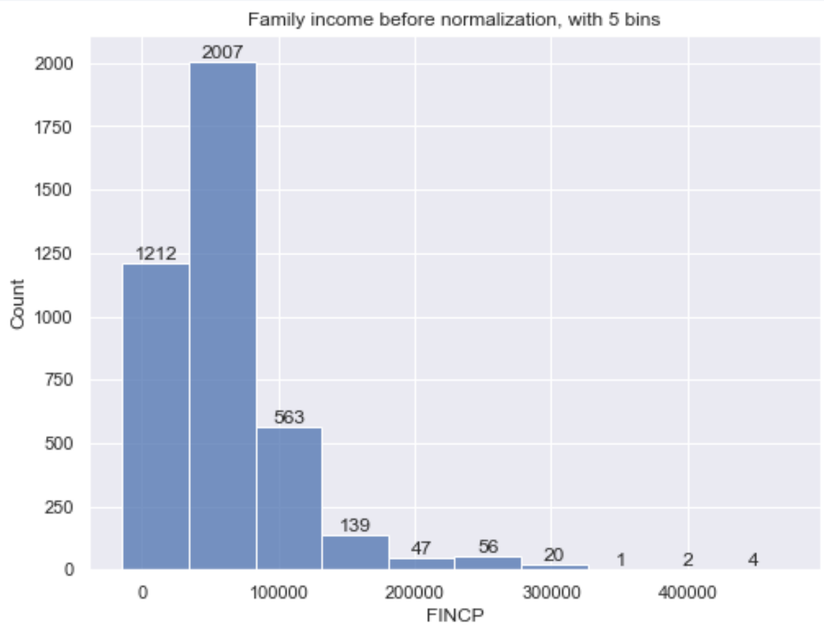
Using NP and VAL together as a key, the count of their occurrence can be calculated and plot with a hue. The darker shade of red represents a higher number of occurrences. **This plot suggests that houses with 2 recorded people living are the most common and are usually in the very broad range of $ 90000 to $249999**

Scatterplot with density (joint plot)

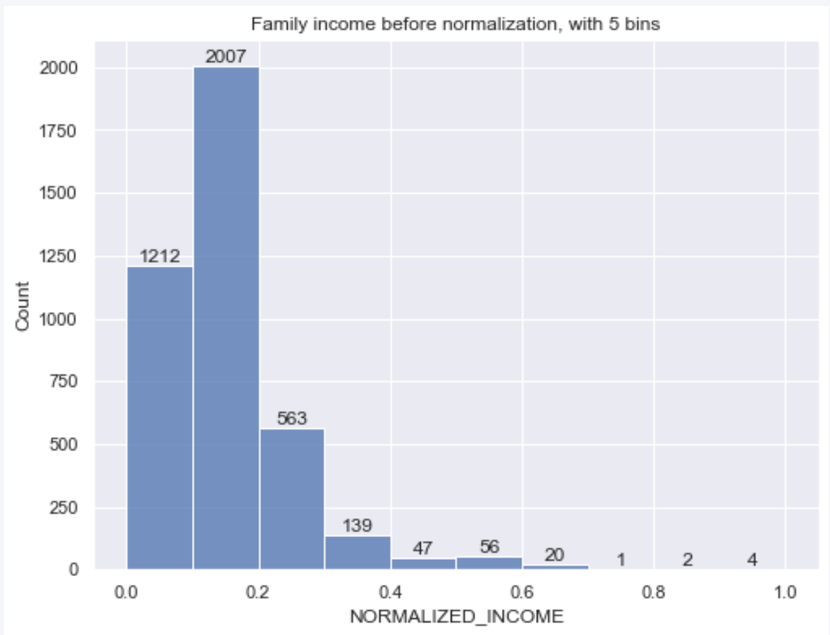
Using Seaborn's joint grid plot, the same result is shown accompanied by the histogram of both columns, where the brighter regions signify the higher density of the NP-VAL pairs. It can also be observed that it is not common to find houses with more than 6 people living in them until property prices above $60000.

Q4: Normalize family income into a range (0-1). Compare before vs after in histogram

Before normalization



After normalization



Min-max normalization

Doing min-max normalization to values between 0 and 1 ensures that all variables are on the same scale and have the same importance when analyzing or modeling the data

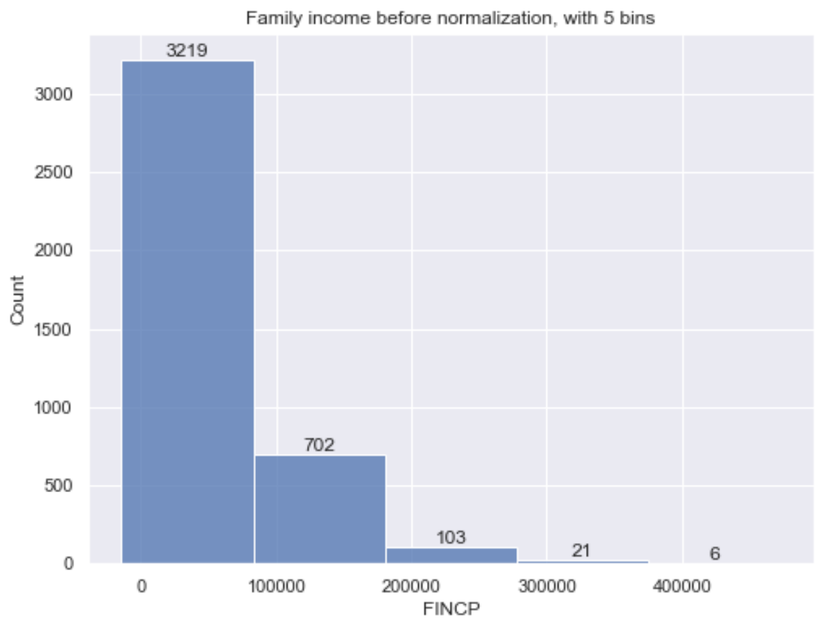
$$x_{scaled} = \frac{x - x_{min}}{x_{max} - x_{min}}$$

It can make it easier to identify patterns and relationships in the data by making it easier to compare variables.

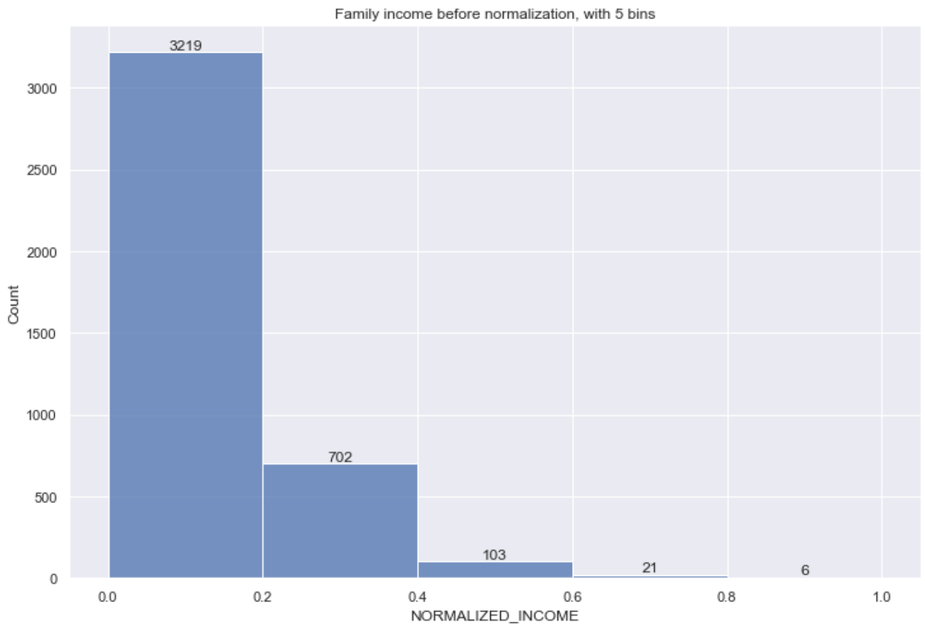
Note that there are **observed negative values in the histogram before normalization** due to the default binning.
Doing a **min-max normalization solves this problem.**

Q5: Create 5 bins for family income

Before normalization



After normalization



Min-max normalization

The data bar plot is then grouped into 5 bins. The left plot shows the data before min-max normalization, and the right shows the data after normalization.

$$x_{scaled} = \frac{x - x_{min}}{x_{max} - x_{min}}$$

Once again, the histogram prior to normalization displays negative values, and normalization also solves it in the case with 5 bins.