Required Assignment 5.1: Will the Customer Accept the Coupon?

Context

Imagine driving through town and a coupon is delivered to your cell phone for a restaurant near where you are driving. Would you accept that coupon and take a short detour to the restaurant? Would you accept the coupon but use it on a subsequent trip? Would you ignore the coupon entirely? What if the coupon was for a bar instead of a restaurant? What about a coffee house? Would you accept a bar coupon with a minor passenger in the car? What about if it was just you and your partner in the car? Would weather impact the rate of acceptance? What about the time of day?

Obviously, proximity to the business is a factor on whether the coupon is delivered to the driver or not, but what are the factors that determine whether a driver accepts the coupon once it is delivered to them? How would you determine whether a driver is likely to accept a coupon?

Overview

The goal of this project is to use what you know about visualizations and probability distributions to distinguish between customers who accepted a driving coupon versus those that did not.

Data

This data comes to us from the UCI Machine Learning repository and was collected via a survey on Amazon Mechanical Turk. The survey describes different driving scenarios including the destination, current time, weather, passenger, etc., and then ask the person whether he will accept the coupon if he is the driver. Answers that the user will drive there ‘right away’ or ‘later before the coupon expires’ are labeled as ‘Y = 1’ and answers ‘no, I do not want the coupon’ are labeled as ‘Y = 0’. There are five different types of coupons -- less expensive restaurants (under
20), coffee houses, carry out & take away, bar, and more expensive restaurants (20 - $50).

Deliverables

Your final product should be a brief report that highlights the differences between customers who did and did not accept the coupons. To explore the data you will utilize your knowledge of plotting, statistical summaries, and visualization using Python. You will publish your findings in a public facing github repository as your first portfolio piece.

Data Description

Keep in mind that these values mentioned below are average values.

The attributes of this data set include:

1. User attributes

    - Gender: male, female
    - Age: below 21, 21 to 25, 26 to 30, etc.
    - Marital Status: single, married partner, unmarried partner, or widowed
    - Number of children: 0, 1, or more than 1
    - Education: high school, bachelors degree, associates degree, or graduate degree
    - Occupation: architecture & engineering, business & financial, etc.
    - Annual income: less than $12500, $12500 - $24999, $25000 - $37499, etc.
    - Number of times that he/she goes to a bar: 0, less than 1, 1 to 3, 4 to 8 or greater than 8
    - Number of times that he/she buys takeaway food: 0, less than 1, 1 to 3, 4 to 8 or greater

    than 8

    - Number of times that he/she goes to a coffee house: 0, less than 1, 1 to 3, 4 to 8 or greater than 8

    - Number of times that he/she eats at a restaurant with average expense less than $20 per

    person: 0, less than 1, 1 to 3, 4 to 8 or greater than 8

    - Number of times that he/she goes to a bar: 0, less than 1, 1 to 3, 4 to 8 or greater than 8
2. Contextual attributes

    - Driving destination: home, work, or no urgent destination
    - Location of user, coupon and destination: we provide a map to show the geographical

    location of the user, destination, and the venue, and we mark the distance between each two places with time of driving. The user can see whether the venue is in the same direction as the destination.

    - Weather: sunny, rainy, or snowy
    - Temperature: 30F, 55F, or 80F
    - Time: 10AM, 2PM, or 6PM
    - Passenger: alone, partner, kid(s), or friend(s)
3. Coupon attributes

    - time before it expires: 2 hours or one day

In [458...

```
import matplotlib.pyplot as plt
import seaborn as sns
import pandas as pd
import numpy as np
```

Problems

Use the prompts below to get started with your data analysis.

1. Read in the `coupons.csv` file.

In [461...

```
data = pd.read_csv('data/coupons.csv')
```

In [463...

```
data.head()
```

Out[463...

	destination	passanger	weather	temperature	time	coupon	expiration	gende
0	No Urgent Place	Alone	Sunny	55	2PM	Restaurant(<20)	1d	Femal
1	No Urgent Place	Friend(s)	Sunny	80	10AM	Coffee House	2h	Femal
2	No Urgent Place	Friend(s)	Sunny	80	10AM	Carry out & Take away	2h	Femal
3	No Urgent Place	Friend(s)	Sunny	80	2PM	Coffee House	2h	Femal
4	No Urgent Place	Friend(s)	Sunny	80	2PM	Coffee House	1d	Femal

5 rows × 26 columns

2. Investigate the dataset for missing or problematic data.

In [466...

```
missing_data = data.isnull().sum()
print(missing_data)
data.shape
```

```
destination                 0
passanger                   0
weather                     0
temperature                 0
time                        0
coupon                      0
expiration                  0
gender                      0
age                         0
maritalStatus               0
has_children                0
education                   0
occupation                  0
income                      0
car                     12576
Bar                       107
CoffeeHouse               217
CarryAway                 151
RestaurantLessThan20      130
Restaurant20To50          189
toCoupon_GEQ5min            0
toCoupon_GEQ15min           0
toCoupon_GEQ25min           0
direction_same              0
direction_opp               0
Y                           0
dtype: int64
```

Out[466…

```
(12684, 26)
```

3. Decide what to do about your missing data -- drop, replace, other...

In [470…

```
'''
car                        12576, does not matter. do nothing
Bar                          107 , missing data convert to "never" , data['Bar'].value
CoffeeHouse                  217 , missing data convert to "never"   Number of times t
CarryAway                    151, missing data convert to "never"
RestaurantLessThan20         130, missing data convert to "never"
Restaurant20To50             189, missing data convert to "never"
'''
#data.sample()
#data['Restaurant20To50'].value_counts()
##data_filled = df.fillna('never')
data_filled=data.copy()
data_filled[['Bar', 'CoffeeHouse','CarryAway','RestaurantLessThan20','Restaurant20T
data_filled.sample(5)
#missing_data2 = data_filled.isnull().sum()
#print(missing_data2)
```

Out[470…

	destination	passanger	weather	temperature	time	coupon	expiration	gen
4929	Home	Alone	Sunny	80	6PM	Coffee House	1d	Fen
6722	Home	Kid(s)	Sunny	55	6PM	Restaurant(20-50)	1d	Fen
1703	No Urgent Place	Friend(s)	Sunny	80	2PM	Coffee House	1d	Fen
9026	No Urgent Place	Friend(s)	Sunny	30	10PM	Carry out & Take away	1d	M
4869	Work	Alone	Sunny	80	7AM	Restaurant(20-50)	2h	M

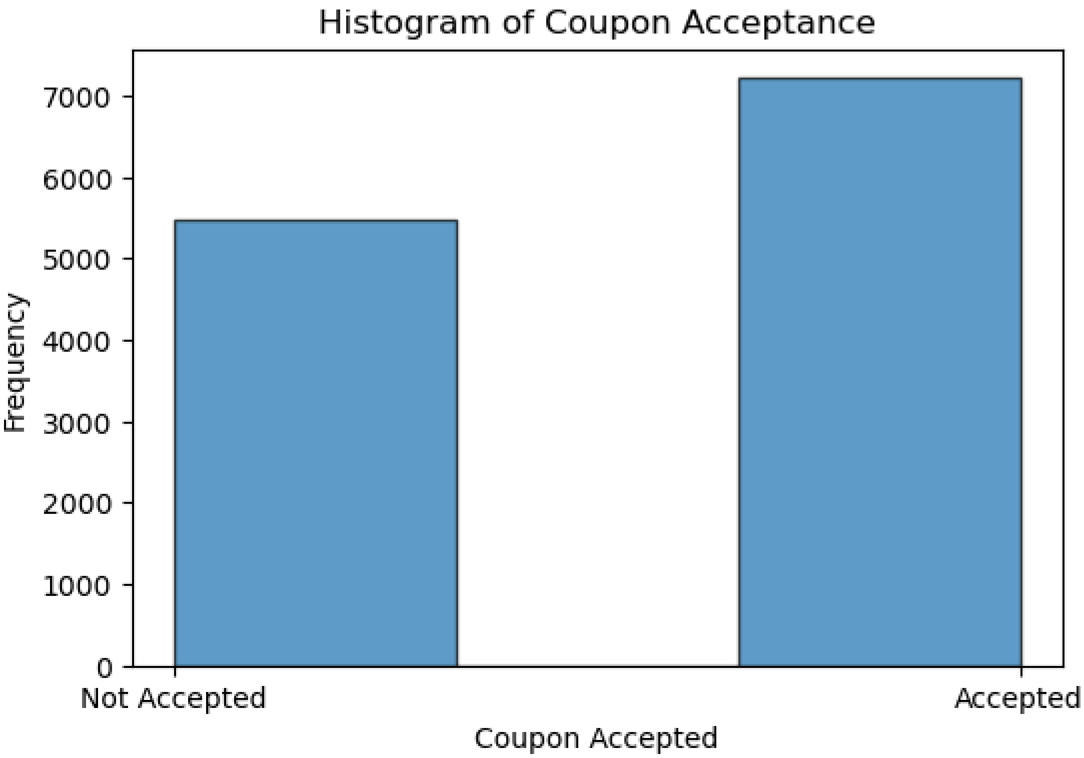
5 rows × 26 columns

4. What proportion of the total observations chose to accept the coupon?

In [472…

```
# Calculate the proportion of observations that chose to accept the coupon
acceptance_percent = data_filled['Y'].mean()

print(f"Percentage of acceptance of the coupon: {acceptance_percent:.2f}")

# Plot the histogram
plt.figure(figsize=(6, 4))
plt.hist(data_filled['Y'], bins=3, edgecolor='black', alpha=0.7)

# Adding labels and title
plt.xlabel('Coupon Accepted')
plt.ylabel('Frequency')
plt.title('Histogram of Coupon Acceptance')

# Show the plot
plt.xticks([0, 1], ['Not Accepted', 'Accepted'])
## the savefig must be placed before the show()
plt.savefig('data/coupon_acceptance_histogram.png', format='png', dpi=600)
plt.show()
```

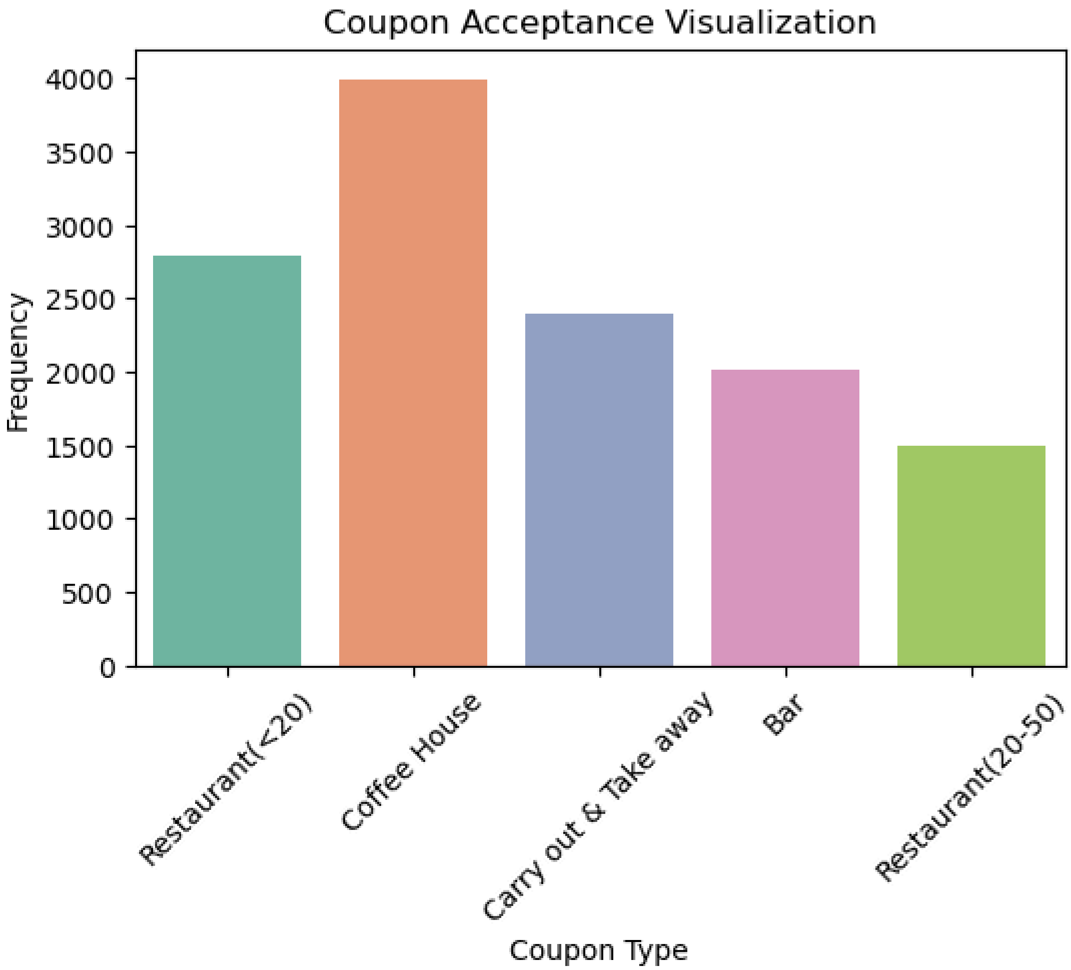
```
Percentage of acceptance of the coupon: 0.57
```



5. Use a bar plot to visualize the `coupon` column.

In [475…

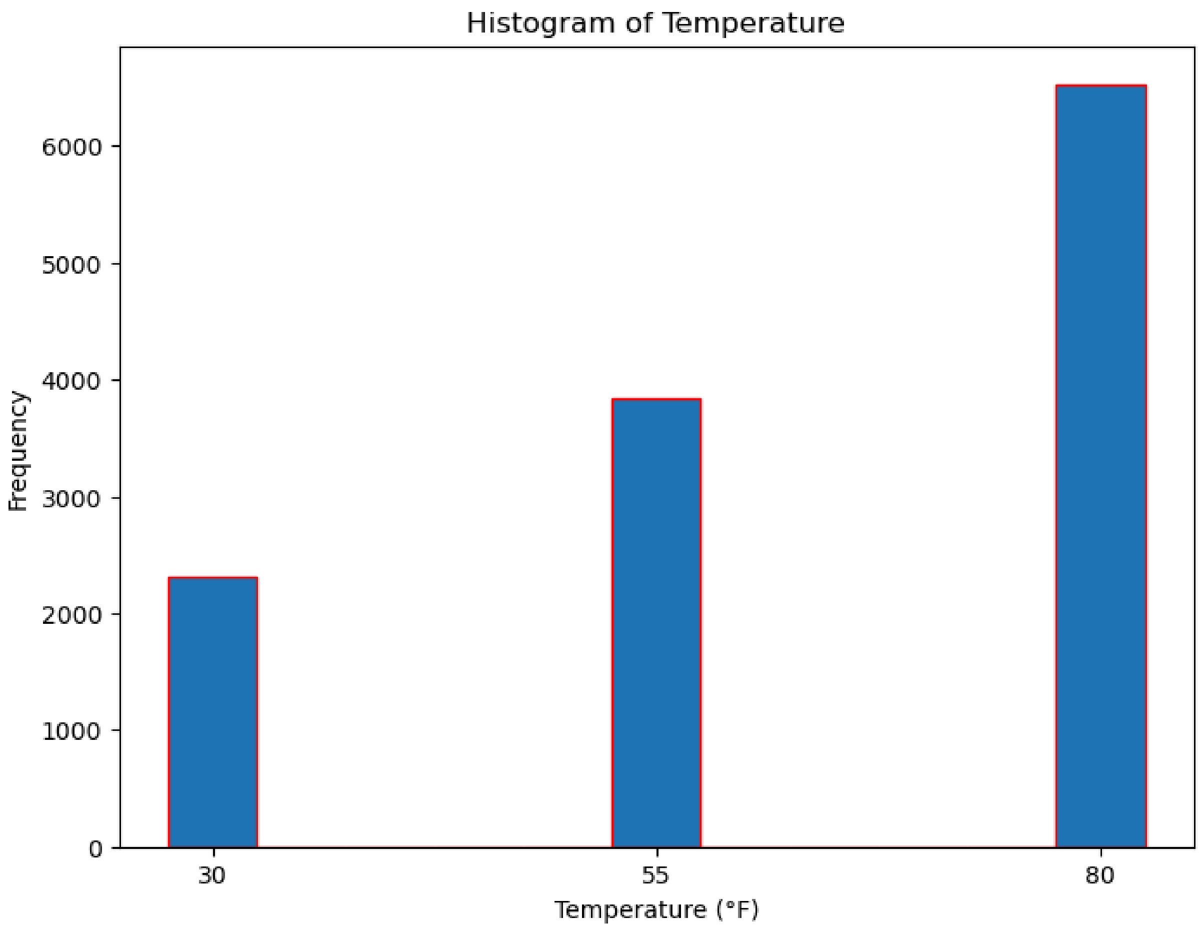
```
# Create a bar plot to visualize the 'coupon' column
plt.figure(figsize=(6, 4))
sns.countplot(x='coupon', data=data_filled, palette='Set2', hue='coupon')

# Adding labels and title
plt.xlabel('Coupon Type')
plt.xticks(rotation=45)
plt.ylabel('Frequency')
plt.title('Coupon Acceptance Visualization')

# Show the plot

plt.savefig('data/coupon_hist.png', format='png', dpi=600)
plt.show()
```



6. Use a histogram to visualize the temperature column.

In [478…

```
# Plotting the histogram using seaborn
plt.figure(figsize=(8, 6))
#sns.histplot(data_filled['temperature'], kde=True, bins=12, color='blue', alpha=0.
bins=range(min(data_filled['temperature']), max(data_filled['temperature']) + 6, 5)
plt.hist(data_filled['temperature'], bins=bins, edgecolor='red',align='left')  # In


# Adding labels and title
plt.xlabel('Temperature (°F)')
plt.ylabel('Frequency')
plt.title('Histogram of Temperature')
# Set x-axis ticks with increments of 25
plt.xticks(range(min(data_filled['temperature']), max(data_filled['temperature'])+5

# Show the plot
plt.savefig('data/temperature_hist.png', format='png', dpi=600)
plt.show()

data_filled['temperature'].value_counts()
#print(bins)
```



Out[478…

```
temperature
80    6528
55    3840
30    2316
Name: count, dtype: int64
```

In [480…

```
list(range(30, 86,5))
```

Out[480…

```
[30, 35, 40, 45, 50, 55, 60, 65, 70, 75, 80, 85]
```

In [ ]:

In [ ]:

In [484…

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 12684 entries, 0 to 12683
Data columns (total 26 columns):
 #   Column                Non-Null Count  Dtype 
---  ------                --------------  ----- 
 0   destination           12684 non-null  object
 1   passanger             12684 non-null  object
 2   weather               12684 non-null  object
 3   temperature           12684 non-null  int64 
 4   time                  12684 non-null  object
 5   coupon                12684 non-null  object
 6   expiration            12684 non-null  object
 7   gender                12684 non-null  object
 8   age                   12684 non-null  object
 9   maritalStatus         12684 non-null  object
 10  has_children          12684 non-null  int64 
 11  education             12684 non-null  object
 12  occupation            12684 non-null  object
 13  income                12684 non-null  object
 14  car                   108 non-null    object
 15  Bar                   12577 non-null  object
 16  CoffeeHouse           12467 non-null  object
 17  CarryAway             12533 non-null  object
 18  RestaurantLessThan20  12554 non-null  object
 19  Restaurant20To50      12495 non-null  object
 20  toCoupon_GEQ5min      12684 non-null  int64 
 21  toCoupon_GEQ15min     12684 non-null  int64 
 22  toCoupon_GEQ25min     12684 non-null  int64 
 23  direction_same        12684 non-null  int64 
 24  direction_opp         12684 non-null  int64 
 25  Y                     12684 non-null  int64 
dtypes: int64(8), object(18)
memory usage: 2.5+ MB
```

Investigating the Bar Coupons

Now, we will lead you through an exploration of just the bar related coupons.

1. Create a new `DataFrame` that contains just the bar coupons.

In [491…

```
data_bar=data_filled.query('coupon=="Bar"')
#data_bar.shape
data_bar.sample(5)
```

Out[491…

	destination	passanger	weather	temperature	time	coupon	expiration	gender
8702	No Urgent Place	Partner	Sunny	30	10AM	Bar	1d	Male
10951	No Urgent Place	Friend(s)	Sunny	80	10PM	Bar	1d	Female
12067	No Urgent Place	Friend(s)	Sunny	80	10PM	Bar	1d	Female
7005	No Urgent Place	Friend(s)	Rainy	55	10PM	Bar	2h	Female
12175	No Urgent Place	Friend(s)	Sunny	55	10PM	Bar	2h	Male

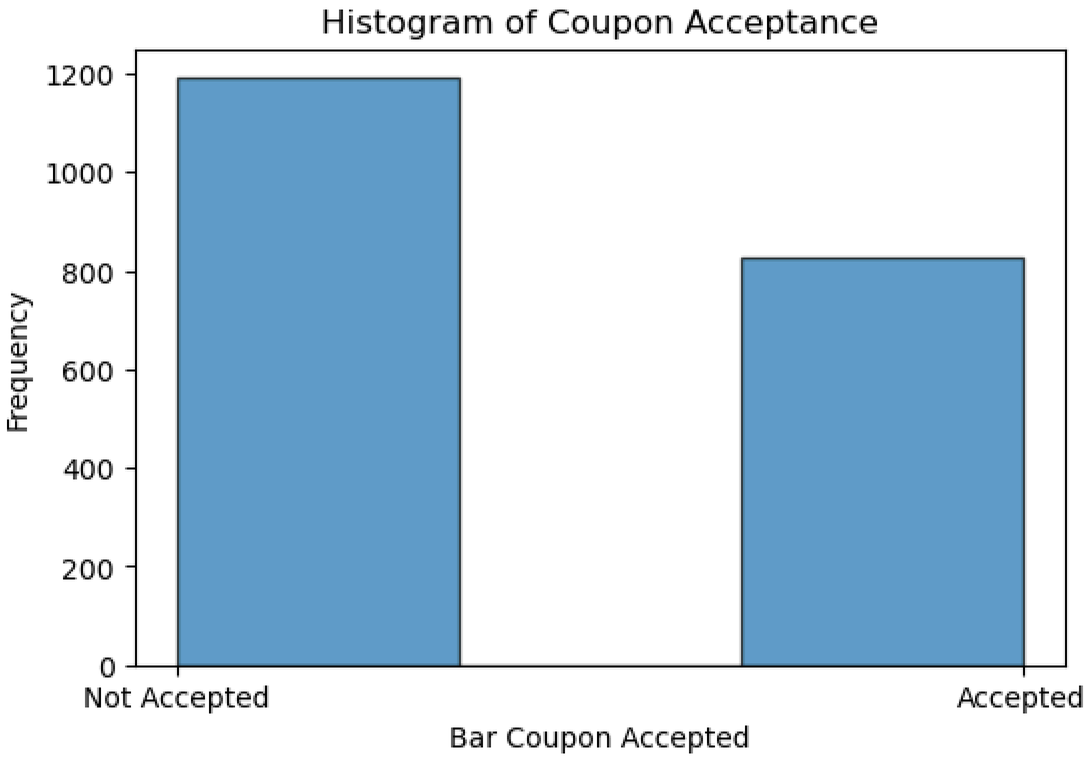
5 rows × 26 columns

2. What proportion of bar coupons were accepted?

In [494…

```python
# Calculate the proportion of observations that chose to accept the bar coupon
acceptance_percent = data_bar['Y'].mean()

print(f"Percentage of acceptance of the coupon: {acceptance_percent:.2f}")

# Plot the histogram
plt.figure(figsize=(6, 4))
plt.hist(data_bar['Y'], bins=3, edgecolor='black', alpha=0.7)

# Adding labels and title
plt.xlabel('Bar Coupon Accepted')
plt.ylabel('Frequency')
plt.title('Histogram of Coupon Acceptance')

# Show the plot
plt.xticks([0, 1], ['Not Accepted', 'Accepted'])
## the savefig must be placed before the show()
plt.savefig('data/Bar_coupon_acceptance_histogram.png', format='png', dpi=600)
plt.show()
```

```
Percentage of acceptance of the coupon: 0.41
```

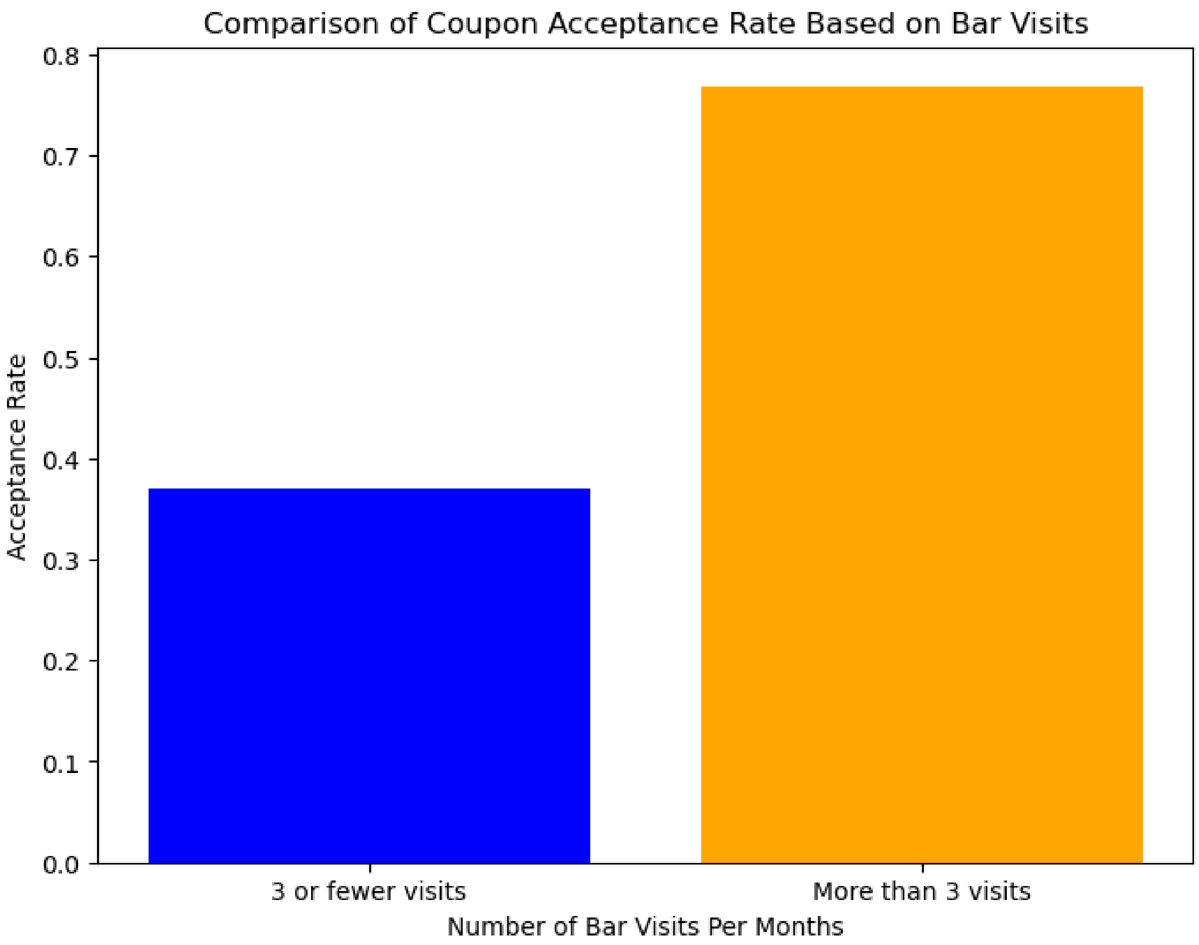


3. Compare the acceptance rate between those who went to a bar 3 or fewer times a month to those who went more.

In [497…

```
data_bar[data_bar['Bar'].isin ( ['never','less1','1~3'])]
Bar_acceptance_3less = data_bar[data_bar['Bar'].isin ( ['never','less1','1~3'])]['Y
Bar_acceptance_3more = data_bar[data_bar['Bar'].isin ( ['4~8','gt8'])]['Y'].mean()

print(f"Percentage of bar acceptance for people who went to a bar 3 or fewer a mont
print(f"Percentage of acceptance people who went to a bar more than 3 a month: {Bar

# Create a bar plot to visualize the acceptance rates
plt.figure(figsize=(8, 6))
plt.bar(['3 or fewer visits', 'More than 3 visits'], [Bar_acceptance_3less, Bar_acc

# Add labels and title
plt.xlabel('Number of Bar Visits Per Months')
plt.ylabel('Acceptance Rate')
plt.title('Comparison of Coupon Acceptance Rate Based on Bar Visits')

# Show the plot
plt.savefig('data/Bar_coupon_acceptance_rate.png', format='png', dpi=600)
plt.show()
```

```
Percentage of bar acceptance for people who went to a bar 3 or fewer a month: 0.37
Percentage of acceptance people who went to a bar more than 3 a month: 0.77
```

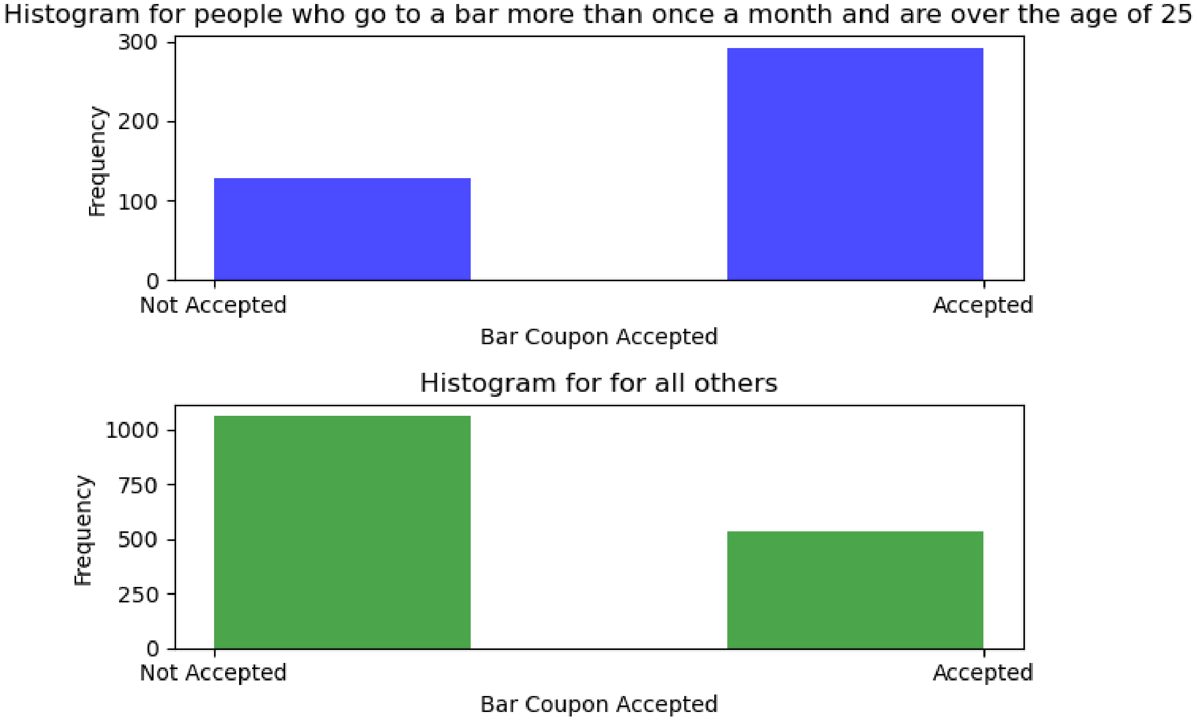


4. Compare the acceptance rate between drivers who go to a bar more than once a month and are over the age of 25 to the all others. Is there a difference?

In [500…

```
data_filled['age']=data_filled['age'].replace({'50plus': '50', 'below21': '21'})
#data_filled['age'] = pd.to_numeric(data_filled['age'], errors='coerce')
data_filled['age'] = data_filled['age'].astype(int)

data_bar=data_filled.query('coupon=="Bar"')

Bar_acceptance_overall = data_bar['Y'].mean()
Bar_frequency=['1~3','4~8','gt8']
person_count_bar_all=data_bar.shape[0]
person_count_1more_over25=data_bar.query('(Bar in @Bar_frequency) and (age > 25)').
person_count_other=data_bar.query('(Bar not in @Bar_frequency) or (age <= 25)').sha
Bar_acceptance_1more_over25 = data_bar.query('(Bar in @Bar_frequency) and (age > 25
Bar_acceptance_others = data_bar.query('(Bar not in @Bar_frequency) | (age <= 25)')

print(f"Percentage of bar acceptance for all ({person_count_bar_all} person): {Bar_
print(f"Percentage of bar acceptance for 1more and 25+ years ({person_count_1more_o
print(f"Percentage of bar acceptance for others ({person_count_other} person):: {Ba

# Plot for Condition 1
plt.subplot(2, 1, 1)
plt.hist(data_bar.query('(Bar in @Bar_frequency) and (age > 25)')['Y'], bins=3, col
plt.title('Histogram for people who go to a bar more than once a month and are over
plt.xlabel('Bar Coupon Accepted')
plt.ylabel('Frequency')
plt.xticks([0, 1], ['Not Accepted', 'Accepted'])

# Plot for Condition 2
plt.subplot(2, 1, 2)
plt.hist(data_bar.query('(Bar not in @Bar_frequency) | (age <= 25)')['Y'], bins=3,
plt.title('Histogram for for all others')
plt.xlabel('Bar Coupon Accepted')
plt.ylabel('Frequency')
plt.xticks([0, 1], ['Not Accepted', 'Accepted'])

# Adjust layout
plt.tight_layout()
plt.show()


#data_bar['age'].value_counts()
#data_bar['Bar'].value_counts()
#data_bar.info()
```

```
Percentage of bar acceptance for all (2017 person): 0.41
Percentage of bar acceptance for 1more and 25+ years (420 person): 0.70
Percentage of bar acceptance for others (1597 person):: 0.34
```

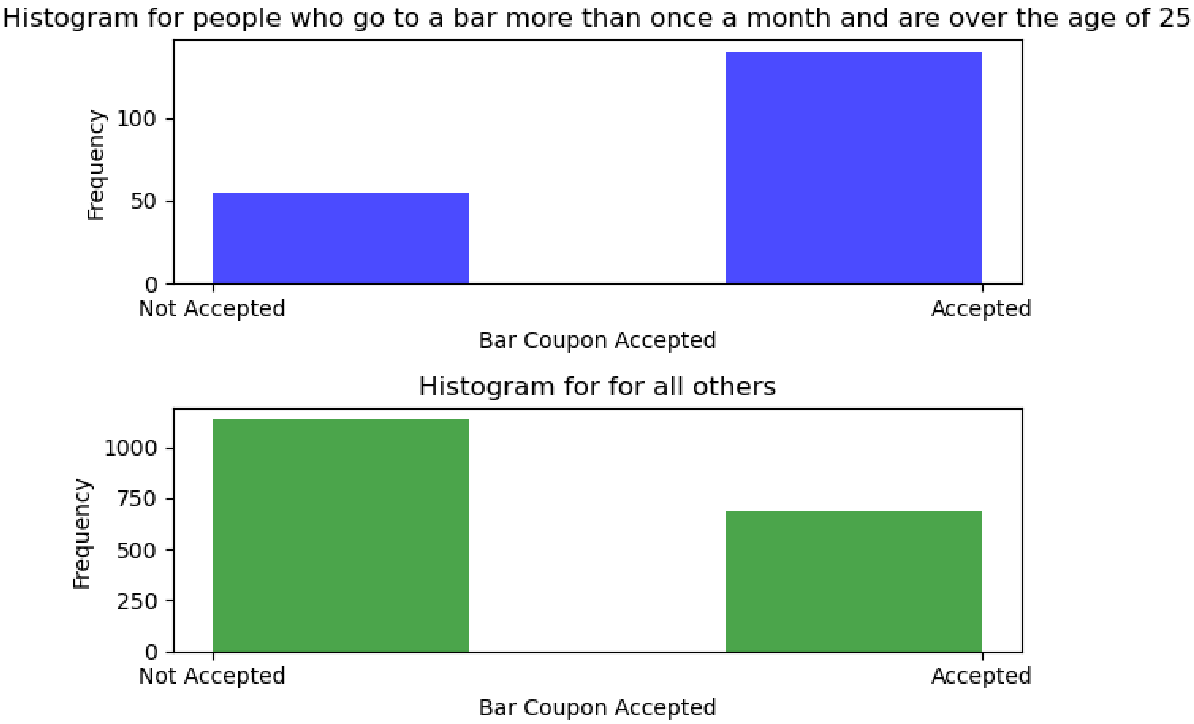


5. Use the same process to compare the acceptance rate between drivers who go to bars more than once a month and had passengers that were not a kid and had occupations other than farming, fishing, or forestry.

```
In [503…
#data_filled['passanger'] = data_filled['passanger'].astype(str)
#data_filled['occupation'] = data_filled['occupation'].astype(str)
data_bar=data_filled.query('coupon=="Bar"')
```

```
Bar_acceptance_overall = data_bar['Y'].mean()
Bar_frequency=['1~3','4~8','gt8']
person_count_bar_all=data_bar.shape[0]
person_count_1more_adultPass=data_bar.query('(Bar in @Bar_frequency) and (passanger
person_count_other=data_bar.query('(Bar not in @Bar_frequency) or (passanger not in
Bar_acceptance_1more_adultPass = data_bar.query('(Bar in @Bar_frequency) and (passa
Bar_acceptance_others = data_bar.query('(Bar not in @Bar_frequency) or (passanger n

print(f"Percentage of bar acceptance for all ({person_count_bar_all} person): {Bar_
print(f"Percentage of bar acceptance for 1more and 25+ years ({person_count_1more_a
print(f"Percentage of bar acceptance for others ({person_count_other} person):: {Ba

# Plot for Condition 1
plt.subplot(2, 1, 1)
plt.hist(data_bar.query('(Bar in @Bar_frequency) and (passanger in ["Friend(s)","Pa
plt.title('Histogram for people who go to a bar more than once a month and are over
plt.xlabel('Bar Coupon Accepted')
plt.ylabel('Frequency')
plt.xticks([0, 1], ['Not Accepted', 'Accepted'])

# Plot for Condition 2
plt.subplot(2, 1, 2)
plt.hist(data_bar.query('(Bar not in @Bar_frequency) or (passanger not in ["Friend(
plt.title('Histogram for for all others')
plt.xlabel('Bar Coupon Accepted')
plt.ylabel('Frequency')
plt.xticks([0, 1], ['Not Accepted', 'Accepted'])

# Adjust layout
plt.tight_layout()
plt.show()

#data_bar.info()
#data_bar['passanger'].value_counts()   #Kid(s) 206 , Alone        1200
#data_bar['occupation'].value_counts()  #Farming Fishing & Forestry 9
```

```
Percentage of bar acceptance for all (2017 person): 0.41
Percentage of bar acceptance for 1more and 25+ years (195 person): 0.72
Percentage of bar acceptance for others (1822 person):: 0.38
```



6. Compare the acceptance rates between those drivers who:

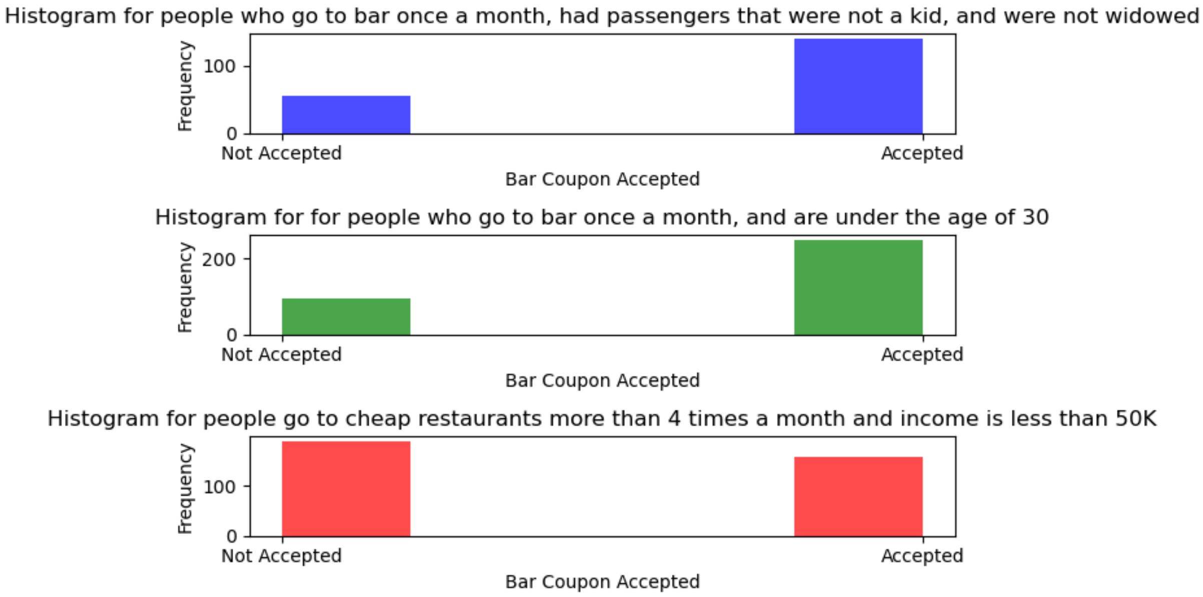
- go to bars more than once a month, had passengers that were not a kid, and were not widowed *OR*
- go to bars more than once a month and are under the age of 30 *OR*
- go to cheap restaurants more than 4 times a month and income is less than 50K.

In [506…

```
data_bar=data_filled.query('coupon=="Bar"')
Bar_frequency=['1~3','4~8','gt8']

# Condition 1: Go to bars more than once, had passengers that are not a kid, and ar
condition_1 = (data_bar['Bar'].isin(Bar_frequency) ) & (data_bar['passanger'].isin(

# Condition 2: Go to bars more than once and are under the age of 30
condition_2 = (data_bar['Bar'].isin(Bar_frequency)) & (data_bar['age'] < 30)

#Condition 3: Go to cheap restaurants more than 4 times and income is less than 50K
income_range=['Less than $12500','$12500 - $24999','$25000 - $37499','$37500 - $499
condition_3 = (data_bar['RestaurantLessThan20'].isin(["4~8","gt8"])) & (data_bar['i


# Calculate acceptance rate for each condition
acceptance_rate_1 = data_bar[condition_1]['Y'].mean()
acceptance_rate_2 = data_bar[condition_2]['Y'].mean()
acceptance_rate_3 = data_bar[condition_3]['Y'].mean()

print(f"Percentage of bar acceptance for condition 1: {acceptance_rate_1:.2f}")
print(f"Percentage of bar acceptance for condition 2: {acceptance_rate_2:.2f}")
print(f"Percentage of bar acceptance for condition 3: {acceptance_rate_3:.2f}")
```

```
# Plot for Condition 1
plt.subplot(3, 1, 1)
plt.hist(data_bar[condition_1]['Y'], bins=5, color='blue', alpha=0.7)
plt.title('Histogram for people who go to bar once a month, had passengers that wer
plt.xlabel('Bar Coupon Accepted')
plt.ylabel('Frequency')
plt.xticks([0, 1], ['Not Accepted', 'Accepted'])

# Plot for Condition 2
plt.subplot(3, 1, 2)
plt.hist(data_bar[condition_2]['Y'], bins=5, color='green', alpha=0.7)
plt.title('Histogram for for people who go to bar once a month, and are under the a
plt.xlabel('Bar Coupon Accepted')
plt.ylabel('Frequency')
plt.xticks([0, 1], ['Not Accepted', 'Accepted'])

# Plot for Condition 3
plt.subplot(3, 1, 3)
plt.hist(data_bar[condition_3]['Y'], bins=5, color='red', alpha=0.7)
plt.title('Histogram for people go to cheap restaurants more than 4 times a month a
plt.xlabel('Bar Coupon Accepted')
plt.ylabel('Frequency')
plt.xticks([0, 1], ['Not Accepted', 'Accepted'])

# Adjust layout
plt.tight_layout()
plt.show()


#data_bar.info()
#data_bar['passanger'].value_counts()   #Kid(s) 206 , Alone        1200
#data_bar['maritalStatus'].value_counts()  #Widowed               21
#data_bar['age'].value_counts()  #Farming Fishing & Forestry 9
#data_bar['RestaurantLessThan20'].value_counts()  #4~8      568   , gt8      186
#data_bar['income'].value_counts()  # $37500 - $49999     267, $12500 - $24999
```

```
Percentage of bar acceptance for condition 1: 0.72
Percentage of bar acceptance for condition 2: 0.72
Percentage of bar acceptance for condition 3: 0.45
```



Histogram for people who go to bar once a month, had passengers that were not a kid, and were not widowed

Histogram for for people who go to bar once a month, and are under the age of 30

Histogram for people go to cheap restaurants more than 4 times a month and income is less than 50K

7. Based on these observations, what do you hypothesize about drivers who accepted the bar coupons?

In [509…

```
print("According to the analysis, it turns out that\n \
1. The overall Bar coupon acceptance is less than 50%, i.e. 41%.\n \
2. People who go to Bar often will more likely to accept the Bar coupon for both ol
3. People make less than 50K income and go to cheap restaurants often are less like
")
```

```
According to the analysis, it turns out that
 1. The overall Bar coupon acceptance is less than 50%, i.e. 41%.
 2. People who go to Bar often will more likely to accept the Bar coupon for both ol
d and young adults.
 3. People make less than 50K income and go to cheap restaurants often are less like
ly to accept the Bar coupon.
```

Independent Investigation

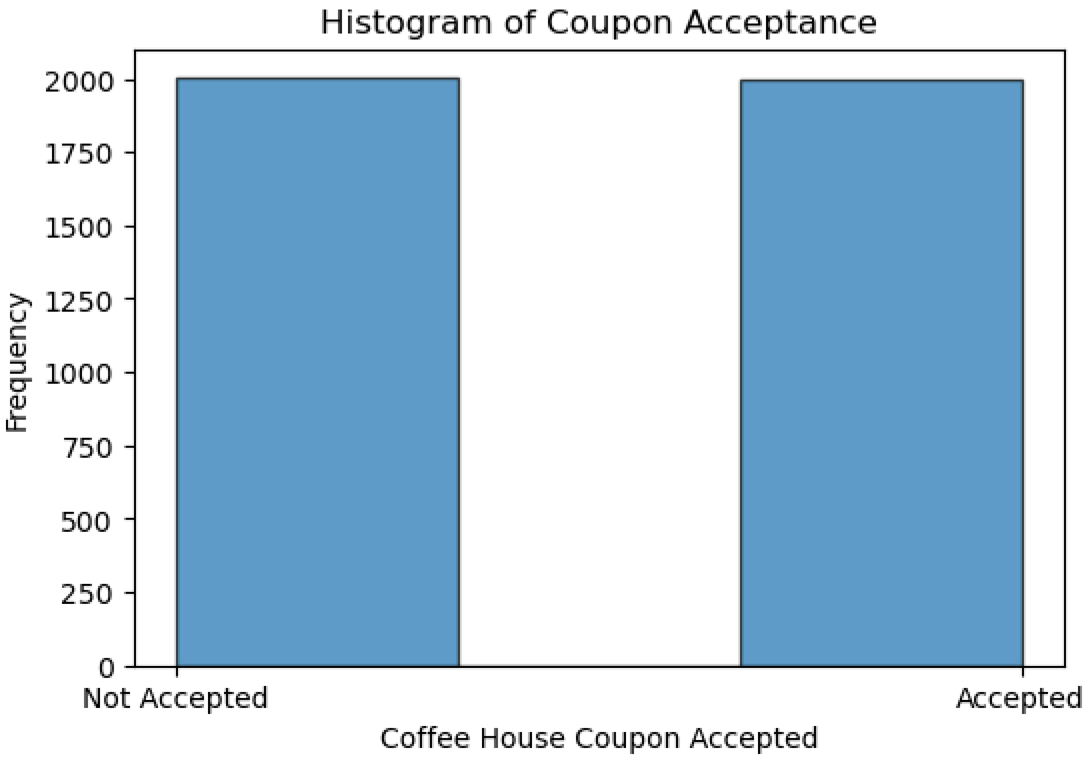
Using the bar coupon example as motivation, you are to explore one of the other coupon groups and try to determine the characteristics of passengers who accept the coupons.

In [511…

```
## investigate the coffee house coupon acceptance because this is related to driver
## overall coffee house coupon acceptance

data_coffee=data_filled.query('coupon=="Coffee House"')
#data_coffee.shape
data_coffee.sample(5)

# Calculate the proportion of observations that chose to accept the bar coupon
acceptance_percent = data_coffee['Y'].mean()

print(f"Percentage of acceptance of the coupon: {acceptance_percent:.2f}")
```

```
# Plot the histogram
plt.figure(figsize=(6, 4))
plt.hist(data_coffee['Y'], bins=3, edgecolor='black', alpha=0.7)

# Adding labels and title
plt.xlabel('Coffee House Coupon Accepted')
plt.ylabel('Frequency')
plt.title('Histogram of Coupon Acceptance')

# Show the plot
plt.xticks([0, 1], ['Not Accepted', 'Accepted'])
plt.show()
```

```
Percentage of acceptance of the coupon: 0.50
```

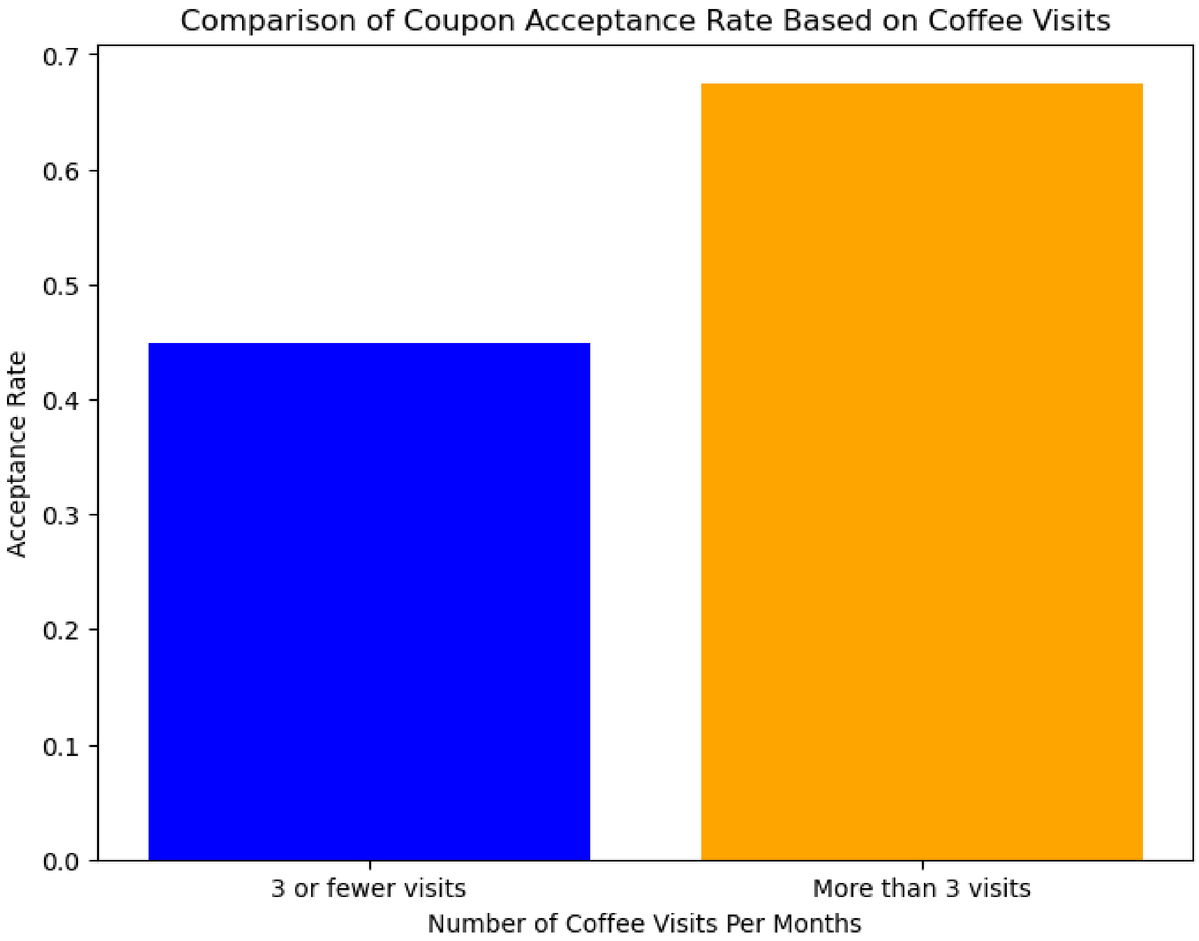


In [523…

```
## coffee house coupon acceptance for these who went to coffee more 3 times a month
data_coffee[data_coffee['CoffeeHouse'].isin( ['never','less1','1~3'])]
Coffee_acceptance_3less = data_coffee[data_coffee['CoffeeHouse'].isin ( ['never','l
Coffee_acceptance_3more = data_coffee[data_coffee['CoffeeHouse'].isin ( ['4~8','gt8

print(f"Percentage of Coffee acceptance for people who went to a Coffee 3 or fewer 
print(f"Percentage of acceptance people who went to a Coffee more than 3 a month: {

# Create a Coffee plot to visualize the acceptance rates
plt.figure(figsize=(8, 6))
plt.bar(['3 or fewer visits', 'More than 3 visits'], [Coffee_acceptance_3less, Coff

# Add labels and title
plt.xlabel('Number of Coffee Visits Per Months')
plt.ylabel('Acceptance Rate')
plt.title('Comparison of Coupon Acceptance Rate Based on Coffee Visits')

# Show the plot
```

```
plt.show()

data_coffee['CoffeeHouse'].value_counts()
```

```
Percentage of Coffee acceptance for people who went to a Coffee 3 or fewer a month:
0.45
Percentage of acceptance people who went to a Coffee more than 3 a month: 0.68
```



```
Out[523…  CoffeeHouse
          less1    1075
          1~3      1042
          never     999
          4~8       538
          gt8       342
          Name: count, dtype: int64
```

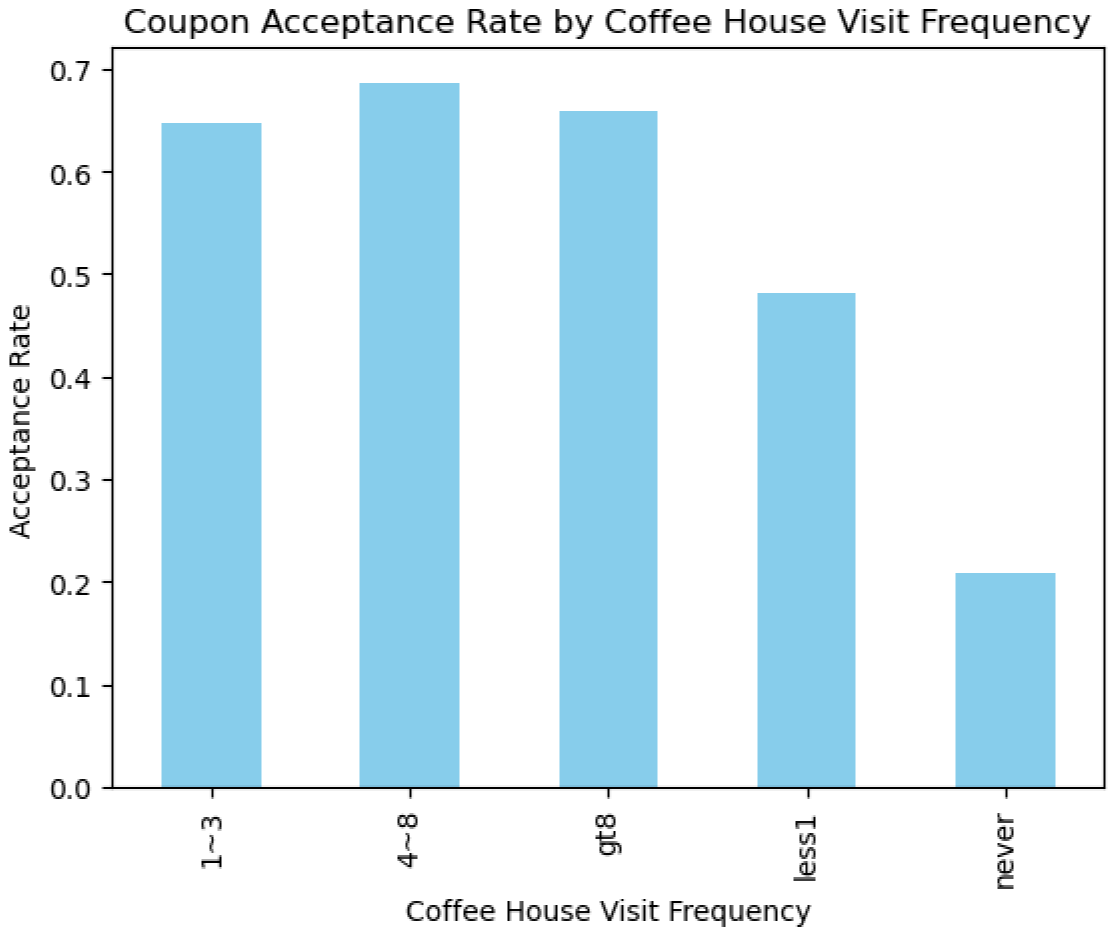
```
In [527…  ## coffee house coupon acceptanceby the group of coffee house visit refrequence

          # Calculate the acceptance rate for each coffee visit frequency
          acceptance_rate_by_coffee = data_coffee.groupby('CoffeeHouse')['Y'].mean()

          # Display the acceptance rates for each group
          print(acceptance_rate_by_coffee)

          # Plotting the acceptance rates
          acceptance_rate_by_coffee.plot(kind='bar', color='skyblue')

          # Adding labels and title
```

```
plt.xlabel('Coffee House Visit Frequency')
plt.ylabel('Acceptance Rate')
plt.title('Coupon Acceptance Rate by Coffee House Visit Frequency')
```

```
CoffeeHouse
1~3      0.647793
4~8      0.685874
gt8      0.657895
less1    0.481860
never    0.208208
Name: Y, dtype: float64
```

Out[527…

```
Text(0.5, 1.0, 'Coupon Acceptance Rate by Coffee House Visit Frequency')
```



In [531…

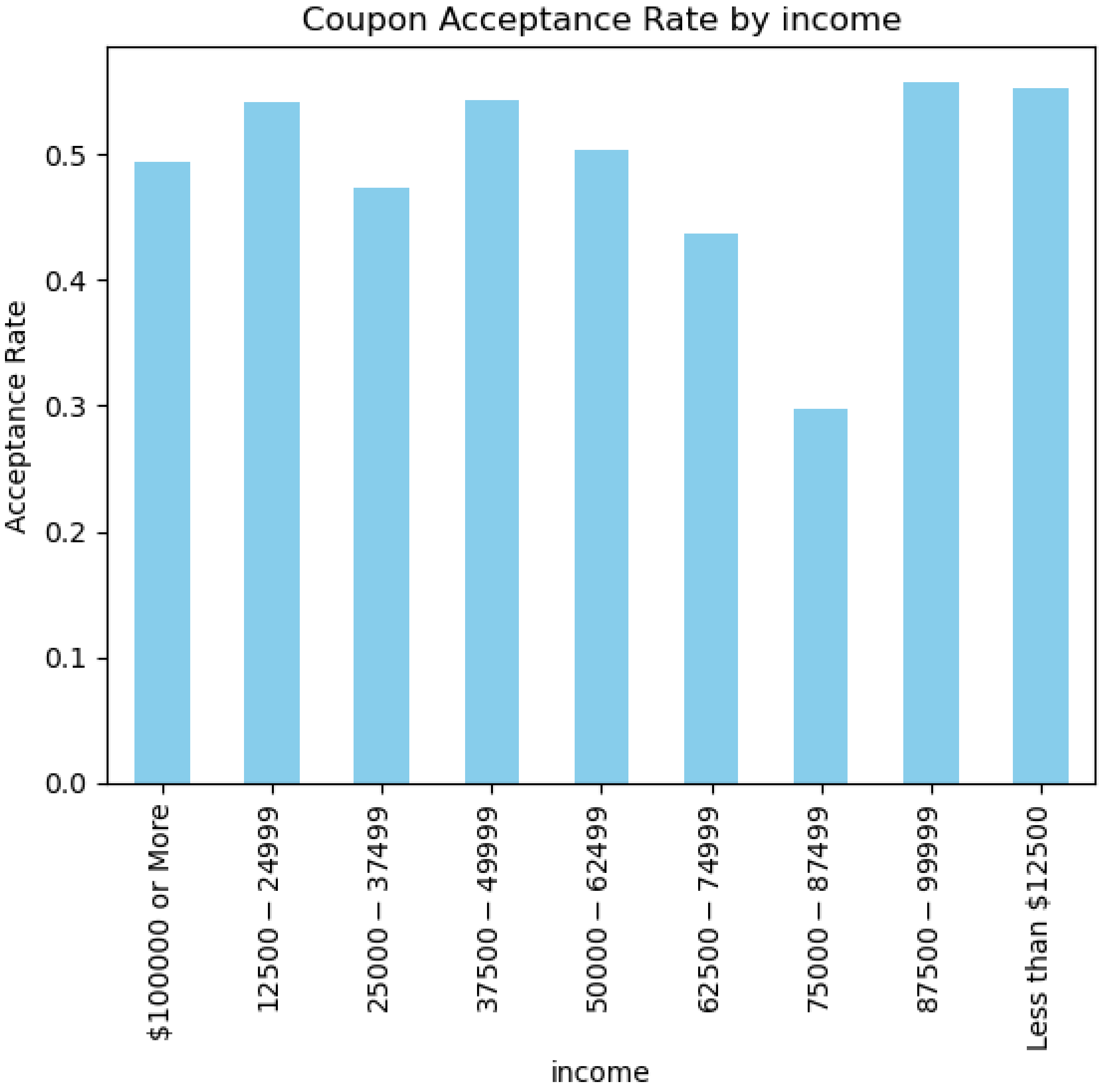
```
## coffee house coupon acceptance by the income
data_Coffee=data_filled.query('coupon=="Coffee"')


# Calculate the acceptance rate for each coffee visit frequency
acceptance_rate_by_income = data_coffee.groupby('income')['Y'].mean()

# Display the acceptance rates for each group
print(acceptance_rate_by_income)

# Plotting the acceptance rates
acceptance_rate_by_income.plot(kind='bar', color='skyblue')

# Adding labels and title
```

```
plt.xlabel('income')
plt.ylabel('Acceptance Rate')
plt.title('Coupon Acceptance Rate by income')
```

```
income
$100000 or More     0.494382
$12500 - $24999     0.540404
$25000 - $37499     0.473438
$37500 - $49999     0.542373
$50000 - $62499     0.503650
$62500 - $74999     0.436364
$75000 - $87499     0.296610
$87500 - $99999     0.557196
Less than $12500    0.551948
Name: Y, dtype: float64
```

Out[531…

```
Text(0.5, 1.0, 'Coupon Acceptance Rate by income')
```



In [533…

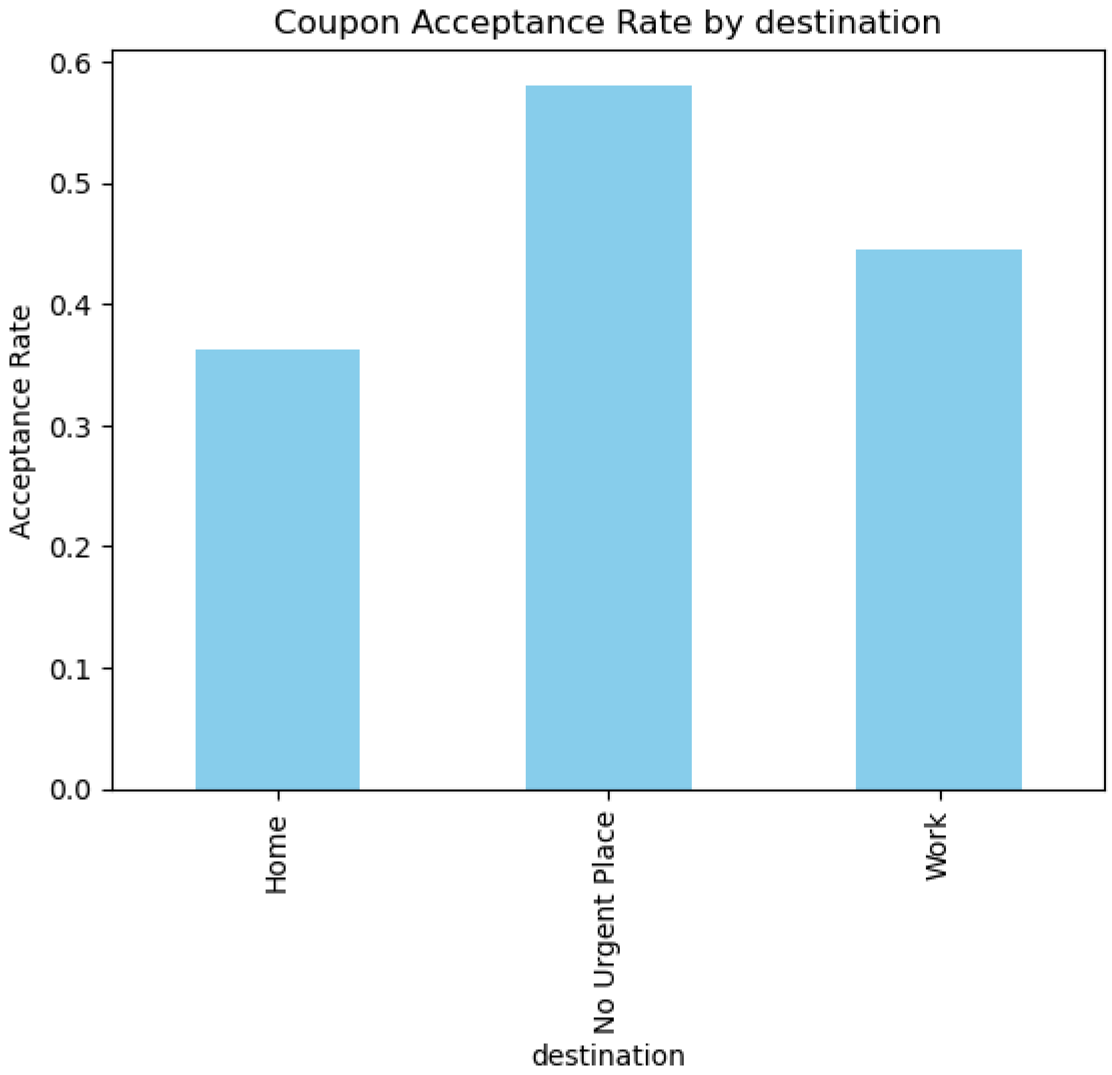
```
## coffee house coupon acceptance for people with different desitnations

# Calculate the acceptance rate for each coffee visit frequency
acceptance_rate_by_destination = data_coffee.groupby('destination')['Y'].mean()
```

```
# Display the acceptance rates for each group
print(acceptance_rate_by_destination)

# Plotting the acceptance rates
acceptance_rate_by_destination.plot(kind='bar', color='skyblue')

# Adding labels and title
plt.xlabel('destination')
plt.ylabel('Acceptance Rate')
plt.title('Coupon Acceptance Rate by destination')
```

```
destination
Home               0.362069
No Urgent Place    0.580974
Work               0.445783
Name: Y, dtype: float64
```

Out[533…

```
Text(0.5, 1.0, 'Coupon Acceptance Rate by destination')
```



In [535…

```
## conclusions
print("According to the analysis, it turns out that\n \
1. The overall Coffee coupon acceptance is 50%, a little interesting. \n \
2. People who go to Coffee House often will more likely to accept the Coffee coupon
3. A surprise that the chance to accept the coffee coupon is about the same for dif
")
```

```
According to the analysis, it turns out that
 1. The overall Coffee coupon acceptance is 50%, a little interesting.
 2. People who go to Coffee House often will more likely to accept the Coffee coupo
n. The chance is 50% if that person go to coffee house once a month.
 3. A surprise that the chance to accept the coffee coupon is about the same for dif
ferent income, especially if making less than $62499. Maybe due to coffee is not exp
ensive.
```

```
In [ ]:
```