Week 1:

I chose the NESARC study dataset. After review of the content, I decided to take a deeper look on drinking behaviour. In detail I would like to analyze, type of beverage consumed (beer or wine) and the correlation to personal income. Further, I would like to know how drinking behaviour variies by age and sex.

For this the data needs to be filtered accordingly to only have the keys "S1Q12B" for "TOTAL HOUSEHOLD INCOME IN LAST 12 MONTHS: CAT", "AGE" for age, "SEX" for sex, as well as the keys for beer and wine consumption, "S2AQ5B" or "S2AQ6B" for how often beer or wine was consumed in last 12 month, as well as "S2AQ5D" "S2AQ6D" for number of beers or wine drank when drinking in the last 12 month (refer to codebook of the NESARC study dataset).

```
206-207 S1Q12B TOTAL HOUSEHOLD INCOME IN LAST 12 MONTHS: CATEGORY

        1531 1. Less than $5,000
    2212 2. $5,000 to $7,999
    1304 3. $8,000 to $9,999
    2437 4. $10,000 to $12,999
    1288 5. $13,000 to $14,999
    3232 6. $15,000 to $19,999
    3326 7. $20,000 to $24,999
    2961 8. $25,000 to $29,999
    3050 9. $30,000 to $34,999
    2605 10. $35,000 to $39,999
    4407 11. $40,000 to $49,999
    3552 12. $50,000 to $59,999
    2729 13. $60,000 to $69,999
    2084 14. $70,000 to $79,999
    1430 15. $80,000 to $89,999
    1011 16. $90,000 to $99,999
    1171 17. $100,000 to $109,999
    451 18. $110,000 to $119,999
    939 19. $120,000 to $149,999
    745 20. $150,000 to 199,999
        628 21. $200,000 or more
```

```
68-69 AGE

    43079 18-97. Age in years
    14 98. 98 years or older
```

```
79-79 SEX SEX
```

18518 1. Male
24575 2. Female

338-339 S2AQ5B HOW OFTEN DRANK BEER IN LAST 12 MONTHS

836 1. Every day
645 2. Nearly every day
1535 3. 3 to 4 times a week
2190 4. 2 times a week
2451 5. Once a week
2603 6. 2 to 3 times a month
2127 7. Once a month
1194 8. 7 to 11 times in the last year
2268 9. 3 to 6 times in the last year
2442 10. 1 or 2 times in the last year
55 99. Unknown
24747 BL. NA, did not drink or unknown if drank beer in last 12 months

---

358-359 S2AQ6B HOW OFTEN DRANK WINE IN LAST 12 MONTHS

465 1. Every day
314 2. Nearly every day
643 3. 3 to 4 times a week
828 4. 2 times a week
1193 5. Once a week
1553 6. 2 to 3 times a month
1819 7. Once a month
1053 8. 7 to 11 times in the last year
2780 9. 3 to 6 times in the last year
3891 10. 1 or 2 times in the last year
22 99. Unknown
28532 BL. NA, did not drink or unknown if drank wine in last 12 months

---

342-343 S2AQ5D NUMBER OF BEERS USUALLY CONSUMED ON DAYS WHEN DRANK BEER IN LAST 12 MONTHS

18268 1-42. Number of beers
78 99. Unknown
24747 BL. NA, did not drink or unknown if drank beer in last 12 months

---

```
362-363 S2AQ6D NUMBER OF GLASSES/CONTAINERS OF WINE USUALLY CONSUMED
ON DAYS WHEN DRANK
```

WINE IN LAST 12 MONTHS

```
    14530 1-12. Number of drinks of wine
    31 99. Unknown
    28532 BL. NA, did not drink or unknown if drank wine in last 12
months
```

I decided to narrow down further, and only focus on wine consumption data and only consider the age and not the sex. Hence, my research questions are:

1. Is there a correlation between drinking wine and the income?
2. Is there a correlation between drinking wine and the age?

To get more information I performed a literature study using keywords like "wine consumption", "age" and "income".

The relationship between drinking behavior, particularly beer and wine consumption, and personal income has been the subject of various studies. Besides others, the following adress the main questions, that I was asking for:

1. Villanueva, Emiliano C.; Castillo-Valero, Juan Sebastián; García-Cortijo, M Carmen: "Who is Drinking Wine in the United States? The Demographic and Socio-Economic Profile of U.S. Wine Consumers (1972-2012), International Food and Agribusiness Management Review, 18, 4:

The study provides a comprehensive demographic and socio-economic profile of wine consumers in the United States over a 40-year period (1972-2012). It concludes that wine consumption has evolved significantly, with notable shifts in the demographics of consumers. The findings indicate that wine drinkers tend to be more affluent, educated, and older compared to non-wine drinkers. Additionally, the research highlights the increasing popularity of wine among younger consumers and women, suggesting a diversification in the wine market and the need for targeted marketing strategies to appeal to these emerging consumer segments.

1. Barber, N., Almanza, B.A. and Donovan, J.R. (2006), "Motivational factors of gender, income and age on selecting a bottle of wine", International Journal of Wine Marketing, Vol. 18 No. 3, pp. 218-232:

This research investigates the motivational factors influencing wine selection based on gender, income, and age. The findings reveal that these demographic factors significantly affect consumer preferences and choices when selecting a bottle of wine. For instance, women are more likely to consider factors such as taste and brand reputation, while men may prioritize price

and alcohol content. Additionally, higher income levels correlate with a preference for premium wines. The study emphasizes the importance of understanding these motivational factors for effective marketing and product positioning in the wine industry.

1. Frank J. Elgar, Chris Roberts, Nina Parry-Langdon, William Boyce, Income inequality and alcohol use: a multilevel analysis of drinking and drunkenness in adolescents in 34 countries, European Journal of Public Health, Volume 15, Issue 3, June 2005, Pages 245–250:

This paper examines the relationship between income inequality and alcohol use among adolescents across 34 countries. The findings indicate that higher levels of income inequality are associated with increased alcohol consumption and drunkenness among adolescents. The study suggests that socio-economic factors play a critical role in shaping drinking behaviors, with adolescents in more unequal societies exhibiting higher rates of risky drinking. The authors advocate for public health interventions that address income inequality as a means to reduce alcohol-related harm among youth.

The literature indicates, that there is a tendency of drinking wine with increasing income. Also, there might be a variation in age observed, due to increasing interest of younger consumers in future.

Based on this, I derive two hypotheses:

H1: "Drinking of wine increases with increasing income"

H2: "The amount of consumed wine is independent of age"

Week 2:

I have been using python for many years in my daily work and this is why I chose this approach to analyze my data.

In the following, I will do the data processing to ensure properly filtered and cleaned data.

In [ ]:
```
import pandas as pd

# Import the raw data as pandas dataframe
rawdata = pd.read_csv('NESARC.csv')

# Have a look on the data to understand the structure
print(rawdata)
```

The data matches the description in the codebook of NESARC database:

```
   ETHRACE2A ETOTLCA2  IDNUM    PSU  STRATUM        WEIGHT  CDAY  CMON  \
0          5                    1   4007      403   3928.613505    14
8
1          5   0.0014       2   6045      604   3638.691845    12
1
2          5                3  12042     1218   5779.032025    23
```

```
11
3              5              4  17099     1704   1071.754303     9
9
4              2              5  17099     1704   4986.952377    18
10
...          ...      ...    ...    ...      ...           ...   ...
...
43088          1          43089  12010     1208  10477.240840    27
11
43089          1   0.2237 43090  17099     1704   9014.746280    30
10
43090          1   0.3785 43091  18094     1802   8079.917091    16
10
43091          1  14.0831 43092  31035     3104  10367.259020    26
9
43092          1          43093  17099     1704   9014.746280     1
11

   CYEAR  REGION  ...  SOL12ABDEP  SOLP12ABDEP  HAL12ABDEP
HALP12ABDEP  \
0       2001       4  ...           0            0           0
0
1       2002       4  ...           0            0           0
0
2       2001       3  ...           0            0           0
0
3       2001       2  ...           0            0           0
0
4       2001       2  ...           0            0           0
0
...      ...     ...  ...         ...          ...         ...
...
43088   2001       3  ...           0            0           0
0
43089   2001       2  ...           0            0           0
0
43090   2001       2  ...           0            0           0
0
43091   2001       2  ...           0            3           0
3
43092   2001       2  ...           0            0           0
0

   MAR12ABDEP  MARP12ABDEP  HER12ABDEP  HERP12ABDEP  OTHB12ABDEP  \
0              0            0           0            0            0
1              0            0           0            0            0
2              0            0           0            0            0
3              0            0           0            0            0
4              0            0           0            0            0
...          ...          ...         ...          ...          ...
43088          0            0           0            0            0
43089          0            0           0            0            0
```

```
43090           0            0           0           0           0
43091           0            3           0           0           0
43092           1            1           0           0           0

      OTHBP12ABDEP
0                    0
1                    0
2                    0
3                    0
4                    0
...                ...
43088                0
43089                0
43090                0
43091                0
43092                0
```

[43093 rows x 3008 columns]

To further filter the data according to my research question, the following is done:

In [ ]:

```
# Apply filter accordingly
filtered_data = rawdata[["AGE", "SEX", "S1Q12B", "S2AQ6B", "S2AQ6D"]].copy()

### Rename the columns for easier identification
# Define a dictionary for renaming columns to have them more meaningful for further p
rename_dict = {
    "S1Q12B": "householdincome",
    "AGE": "age",
    "SEX": "sex",
    "S2AQ6B": "howoftenwine",
    "S2AQ6D": "noofwines"
}

# Rename the columns accordingly, using the dictionary above
filtered_data.rename(columns = rename_dict, inplace=True)

# Print the DataFrame after dropping rows with empty entries
print(filtered_data)
```

I figured out, that there are some blanks in the dataset. Further, there are NaN values to be corrected. I decided to replace them by "999", to clearly distinguish them from other values in the table.

In [ ]:

```
# Replace blank values with "999", to make sure there are no blank values in the data
filtered_data.replace("", "999", inplace=True)

# Replace NaN values with "999", to make sure there are no NaN values in the dataset
filtered_data.fillna("999", inplace=True)

# Print the data frame again
print(filtered_data)
```

The result looks good.

	age	sex	householdincome	howoftenwine	noofwines
0	23	1	11	999	999
1	28	2	10	999	999
2	81	2	2	999	999
3	18	1	11	999	999
4	36	1	15	999	999
...	...	...	...	...	...
43088	18	2	1	999	999
43089	19	1	1	999	999
43090	18	1	1	999	999
43091	29	1	2	999	999
43092	18	1	2	999	999

[43093 rows x 5 columns]

Since the 999 values in the columns for wine consumption amount and frequency are not meaningful for further evaluation, I remove them from the dataframe.

In [ ]:

```
# Drop rows where values are 999 in "howoftenwine" and "noofwines"
df_cleaned = filtered_data[~((filtered_data['howoftenwine'] == 999)& (filtered_data[

# Reset the index
df_cleaned.reset_index(drop=True, inplace=True)

print(df_cleaned)
```

The cleaning was successful. There are 14561 rows remaining for further evaluation on wine consumption behaviour in context of my research question.

	age	sex	householdincome	howoftenwine	noofwines
0	34	2	12	10	1
1	84	2	7	6	1
2	29	2	13	10	1
3	68	2	6	5	1
4	54	2	11	9	1
...	...	...	...	...	...
14556	18	2	1	9	1
14557	18	1	1	10	1
14558	51	1	6	6	1

```
14559   21    1                1            10           2
14560   18    2                1            10           1
```

[14561 rows x 5 columns]

After filtering and cleaning the dataset, I now continue with creating the frequency distributions for each variable.

In [ ]:

```
import numpy as np

# Defining the frequency distribution for the 'age' parameter accoridngly
age_freq = df_cleaned['age'].value_counts().sort_index()

# Set options to display all rows and columns, since the dataset is longer than the s
pd.set_option('display.max_rows', None)
pd.set_option('display.max_columns', None)

# Print the distribution accordingly
print(age_freq)
```

The frequency distribution analysis of age shows that the age ranges from 18 to 96. Further, the ages between 20 and 58 are more often represented in the data. To have a better overview, binning of the age data should be done later on.

```
age
18    124
19    159
20    166
21    235
22    220
23    228
24    260
25    254
26    249
27    251
28    251
29    304
30    309
31    294
32    304
33    287
34    322
35    306
36    297
37    382
38    336
39    370
40    391
41    349
42    350
43    314
```

44	304
45	331
46	315
47	301
48	316
49	260
50	276
51	288
52	254
53	263
54	308
55	217
56	211
57	187
58	241
59	180
60	171
61	160
62	147
63	162
64	141
65	151
66	144
67	130
68	128
69	146
70	123
71	129
72	104
73	129
74	124
75	103
76	87
77	91
78	73
79	74
80	84
81	66
82	48
83	62
84	41
85	40
86	28
87	30
88	18
89	21
90	11
91	10
92	8
93	5
94	5
95	1

```
   96      2
```

Next parameter is the income data. Since it is alreday pre-categorized in the database, I only need to decrypt it and match it with the given category names.

In [ ]:

```
# Create a mapping from numeric codes (given by the NESARC codebook) to income catego
income_mapping = {
    1: 'Less than $5,000',
    2: '$5,000 to $7,999',
    3: '$8,000 to $9,999',
    4: '$10,000 to $12,999',
    5: '$13,000 to $14,999',
    6: '$15,000 to $19,999',
    7: '$20,000 to $24,999',
    8: '$25,000 to $29,999',
    9: '$30,000 to $34,999',
    10: '$35,000 to $39,999',
    11: '$40,000 to $49,999',
    12: '$50,000 to $59,999',
    13: '$60,000 to $69,999',
    14: '$70,000 to $79,999',
    15: '$80,000 to $89,999',
    16: '$90,000 to $99,999',
    17: '$100,000 to $109,999',
    18: '$110,000 to $119,999',
    19: '$120,000 to $149,999',
    20: '$150,000 to $199,999',
    21: '$200,000 or more'
}

# Replace numeric codes with category names using the mapping method
df_cleaned['income_category'] = df_cleaned['householdincome'].map(income_mapping)

# Count occurrences of each income category
income_counts = df_cleaned['income_category'].value_counts()

# Create a summary data frame
summary_income = pd.DataFrame(income_counts).reset_index()
summary_income.columns = ['Income Category', 'Count']

# Sort the summary data frame by the income category
summary_income['Income Category'] = pd.Categorical(summary_income['Income Category'],
                                                   categories=list(income_mapping.values
                                                   ordered=True)

# Sort the data frame
summary_income = summary_income.sort_values('Income Category')

# Display the summary data frane
print(summary_income.to_string(index=False))
```

The result is as follows:

```
      Income Category  Count
      Less than $5,000    334
  $5,000 to $7,999    326
```

```
$8,000 to $9,999    218
```

$10,000 to 12,999 426 $13,000 to 14,999 256 $15,000 to 19,999 750 $20,000 to 24,999 840 $25,000 to 29,999 811 $30,000 to 34,999 944 $35,000 to 39,999 824 $40,000 to 49,999 1612 $50,000 to 59,999 1493 $60,000 to 69,999 1125 $70,000 to 79,999 1006 $80,000 to 89,999 685 $90,000 to 99,999 528 $100,000 to 109,999 660 $110,000 to 119,999 255 $120,000 to 149,999 559 $150,000 to 199,999 486 $200,000 or more 423

Considering the distribution, the income between 50,000 and 79,999 is the most common in the available data.

Finally, I consider the wine drinking frequency and amount data. It is already pre-categorized and the categories can thus be taken over from the NESARC database

In [ ]:

```
# Create a  data frame for the wine consumption frequency category
wine_frequency_mapping = {
    1: 'Every day',
    2: 'Nearly every day',
    3: '3 to 4 times a week',
    4: '2 times a week',
    5: 'Once a week',
    6: '2 to 3 times a month',
    7: 'Once a month',
    8: '7 to 11 times in the last year',
    9: '3 to 6 times in the last year',
    10: '1 or 2 times in the last year',
    99: 'Unknown'
}

# Replace numeric codes with category names using the mapping
df_cleaned['wine_frequency'] = df_cleaned['howoftenwine'].map(wine_frequency_mapping)

# Count occurrences of each category of wine consumption frequency
wine_frequency_counts = df_cleaned['wine_frequency'].value_counts()

# Create a summary data frame
summary_wine = pd.DataFrame(wine_frequency_counts).reset_index()
summary_wine.columns = ['Wine Drinking Frequency', 'Count']

# Sort the summary data frame by the drinking frequency category
summary_wine['Wine Drinking Frequency'] = pd.Categorical(summary_wine['Wine Drinking
                                                       categories=list(wine_frequency_mappi
                                                       ordered=True)

# Sort the data frame
summary_wine = summary_wine.sort_values('Wine Drinking Frequency')

# Display the summary data frame without the index
print(summary_wine.to_string(index=False))
```

The result of the frequency analysis is as follows:

```
       Wine Drinking Frequency  Count
                   Every day    465
```

```
                    Nearly every day    314
                3 to 4 times a week    643
                     2 times a week    828
                        Once a week   1193
               2 to 3 times a month   1553
                       Once a month   1819
  7 to 11 times in the last year      1053
   3 to 6 times in the last year      2780
   1 or 2 times in the last year      3891
                            Unknown     22
```

There are two maximums of the frequency distribution, one for drinking wine 2 to 3 times a month and the second one for 1 or 2 times last year. Further, the amount of wine consumed per occurrence needs to be analyzed:

In [ ]:

```
# Create a DataFrame for the wine consumption frequency category
wine_amount_mapping = {
    1: 'One glass/ container',
    2: 'Two glasses/ containers',
    3: 'Three glasses/ containers',
    4: 'Four glasses/ containers',
    5: 'Five glasses/ containers',
    6: 'Six glasses/ containers',
    7: 'Seven glasses/ containers',
    8: 'Eight glasses/ containers',
    9: 'Nine glasses/ containers',
    10: 'Ten glasses/ containers',
    11: 'Eleven glasses/ containers',
    12: 'Twelve glasses/ containers',
    99: 'Unknown'
}

# Replace numeric codes with category names using the mapping
df_cleaned['wine_amount'] = df_cleaned['noofwines'].map(wine_amount_mapping)

# Count occurrences of each category of wine consumption frequency
wine_amount_counts = df_cleaned['wine_amount'].value_counts()

# Create a summary data frame
summary_wine_amount = pd.DataFrame(wine_amount_counts).reset_index()
summary_wine_amount.columns = ['Wine Drinking Amount', 'Count']

# Sort the summary data frame by the drinking frequency category
summary_wine_amount['Wine Drinking Amount'] = pd.Categorical(summary_wine_amount['Wi
                                                    categories=list(wine_amount_mapping.
                                                    ordered=True)

# Sort the data frame
summary_wine_amount = summary_wine_amount.sort_values('Wine Drinking Amount')

# Display the summary data frame without the index
print(summary_wine_amount.to_string(index=False))
```

The analysis result of drinking amount is as follows:

Wine Drinking Amount	Count
One glass/ container	9004
Two glasses/ containers	4386
Three glasses/ containers	821
Four glasses/ containers	219
Five glasses/ containers	59
Six glasses/ containers	20
Seven glasses/ containers	6
Eight glasses/ containers	5
Nine glasses/ containers	2
Ten glasses/ containers	7
Twelve glasses/ containers	1
Unknown	31

The amount of wine drank in each occurrence is continuously decreasing, with a maximum of one glass/ container.

In [ ]: