Group1_PS2

February 5, 2024

1 Problem Set 2

[63]:
```
import os
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import scipy.optimize as opt
import seaborn as sns
import random as rand

os.chdir("/Users/arthurjohnson/Library/CloudStorage/
↪OneDrive-UniversityofEdinburgh/Year 4/PNM for Economics/
↪PNM-for-Econ-Group-1")
```

1.1 Exercise 1a:

Are there duplicate households in the data? That is check if there are repeated observations in the unique household identifier variable. How many observations are there in the data?

[64]:
```
File_Path = "/Users/arthurjohnson/Desktop/UNPS_1112_PS2.xls"
df = pd.read_excel(File_Path)

#Check for duplicate variables
duplicates = df.duplicated(['hhid'])

#Count the number of duplicates
print("Count of duplicate hououseholds:", duplicates.sum())
```

```
Count of duplicate hououseholds: 0
```

1.1.1 Exercise 1b:

Present some basic summary statistics for the following variables: head_gender, head _age, familysize, consumption, income, wealth. Comment your results in 2 lines. In particular, you might mention if there are missing observations or potential outliers for some of the variables.

[65]:
```
df_info = df.info()

# Checking which columns have 2613 entries(i.e no missing values) entries
```

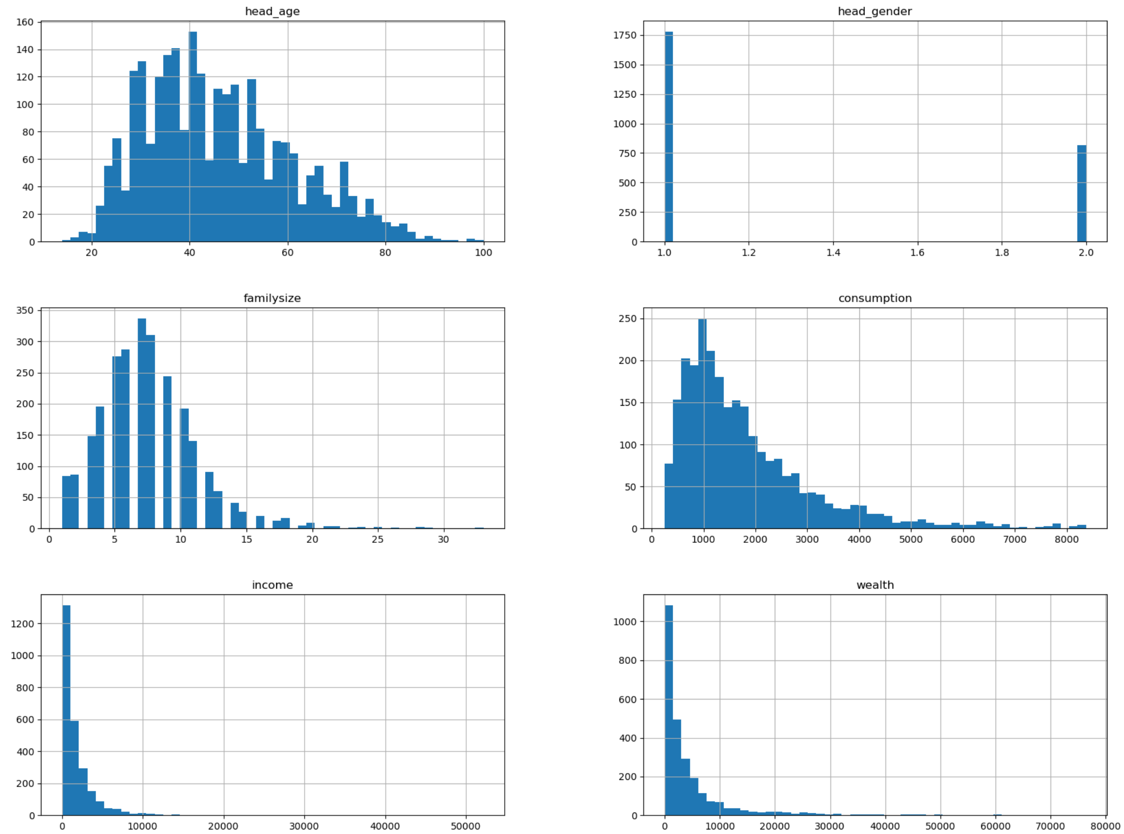
```python
full_columns = df.columns[df.count() == 2613].tolist()
print("Columns with 2613 entries:", full_columns)

#Checking which columns have missing values
miss_columns = df.columns[df.count() < 2613].tolist()
print("Columns with missing values:", miss_columns)
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2613 entries, 0 to 2612
Data columns (total 30 columns):
 #   Column              Non-Null Count  Dtype
---  ------              --------------  -----
 0   hhid                2613 non-null   int64
 1   wave                2613 non-null   object
 2   year                2613 non-null   int64
 3   region              2613 non-null   int64
 4   district            2467 non-null   object
 5   county              2461 non-null   object
 6   urban               2613 non-null   int64
 7   year_surv           2613 non-null   int64
 8   month_surv          2613 non-null   int64
 9   head_gender         2597 non-null   float64
 10  head_age            2597 non-null   float64
 11  head_ethnic         2597 non-null   float64
 12  head_writeread_yes  2597 non-null   float64
 13  head_classeduc      2079 non-null   float64
 14  familysize          2597 non-null   float64
 15  consumption         2613 non-null   float64
 16  cfood               2613 non-null   float64
 17  cnodur              2605 non-null   float64
 18  cdur                2606 non-null   float64
 19  cfood_gift          2613 non-null   float64
 20  income              2613 non-null   float64
 21  wage_labor          896 non-null    float64
 22  business_inc        1110 non-null   float64
 23  other_inc           2569 non-null   float64
 24  agriculture_inc     2090 non-null   float64
 25  livestock_inc       519 non-null    float64
 26  wealth              2613 non-null   float64
 27  asset_value         2554 non-null   float64
 28  wealth_agrls        2054 non-null   float64
 29  land_value_hat      1882 non-null   float64
dtypes: float64(21), int64(6), object(3)
memory usage: 612.6+ KB
Columns with 2613 entries: ['hhid', 'wave', 'year', 'region', 'urban',
'year_surv', 'month_surv', 'consumption', 'cfood', 'cfood_gift', 'income',
'wealth']
Columns with missing values: ['district', 'county', 'head_gender', 'head_age',
```

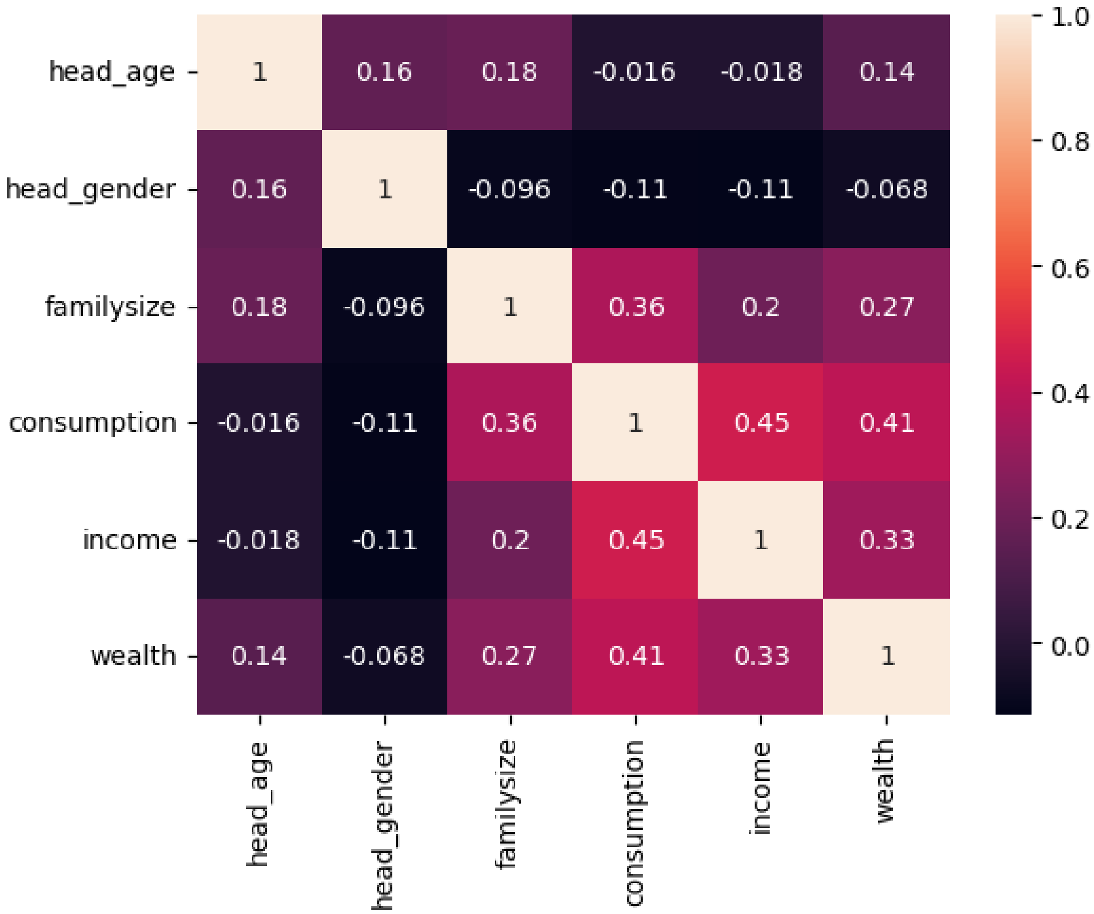
```
'head_ethnic', 'head_writeread_yes', 'head_classeduc', 'familysize', 'cnodur',
'cdur', 'wage_labor', 'business_inc', 'other_inc', 'agriculture_inc',
'livestock_inc', 'asset_value', 'wealth_agrls', 'land_value_hat']
```

[66]:
```python
#Describing specific columns
columns_of_note = ['head_age', 'head_gender', 'familysize', 'consumption',␣
 ↪'income', 'wealth']
description = df[columns_of_note].describe()

# Displaying the results
print(description)

#Plotting histograms of each varaible to get an idea of shape of distributions
df[columns_of_note].hist(bins=50, figsize=(20,15))
plt.show()
```

```
          head_age  head_gender   familysize  consumption        income  \
count  2597.000000  2597.000000  2597.000000  2613.000000   2613.000000
mean     46.068156     1.314209     7.480554  1803.792687   1860.075795
std      15.068960     0.464289     3.712526  1308.742941   2607.517603
min      14.000000     1.000000     1.000000   250.305506     27.486921
25%      34.000000     1.000000     5.000000   918.324958    471.204368
50%      44.000000     1.000000     7.000000  1426.614855   1061.194208
75%      56.000000     2.000000     9.000000  2296.859516   2234.800590
max     100.000000     2.000000    33.000000  8369.898484  52137.736864

             wealth
count   2613.000000
mean    4912.698146
std     8359.886497
min        0.000000
25%      720.019916
50%     2108.598380
75%     5221.939857
max    76396.339917
```



[67]:
```
#Generate correlation matrix
df_matrix = df[columns_of_note].corr()
#Generate heat map to visualise this matrix and show the correlation between the variables of interest.
sns.heatmap(df_matrix, annot=True)
plt.show()
```



We observe that head_age, head_gender, and familysize are all missing exactly 16 values. The variables familysize, wealth, and income have some large outliers. We observe that consumption, wealth, and income are all highly correlated.

1.1.2 Exercise 1c:

Using the head_gender variable, create a dummy variable for household head being female (1=female, 0=male). What is the proportion of households where the head is female?

[68]:
```
# Creating a new dummy variable
df['female_dummy'] = df['head_gender'].apply(lambda x: 1 if x == 2 else 0)

# Counting the number of males and females
male_count = df['female_dummy'].value_counts().get(0, 0)
female_count = df['female_dummy'].value_counts().get(1, 0)

# Displaying the counts
print("Number of males:", male_count)
print("Number of females:", female_count)
```

```
# Calculating the proportion
total_count = len(df)
male_proportion = male_count / total_count
female_proportion = female_count / total_count

# Displaying the proportions
print("Proportion of males:", round(male_proportion, 3))
print("Proportion of females:", round(female_proportion, 3))
```

```
Number of males: 1797
Number of females: 816
Proportion of males: 0.688
Proportion of females: 0.312
```

1.1.3 Exercise 1d:

Using the groupby method, compute the average consumption, average household size, and average household head age for households where the head is male vs where the head is female. Do we observe noticeable differences across the two groups?

[69]:
```
#Group our gender household variable by the value of the dummy
grouped_means = df.groupby('female_dummy')[['consumption', 'familysize',␣
↪'head_age']].mean()

print(grouped_means)
```

```
              consumption  familysize   head_age
female_dummy
0             1899.392959    7.720943  44.388546
1             1593.261206    6.955882  49.734069
```

We notice a difference in level of consumption, with households where the head is male having greater overall levels of consumption. The difference as a proportion is as follows:

$$\frac{male\ consumption}{female\ consumption} = \frac{1899.392959}{1593.261206} \approx 1.1921$$

We postulate that this could be due to the reduced earning potential of women in Ugandan society, leading to households led by women having a lower overall household income and therefore lower consumption. Due to gender norms within Uganda, many female led households will be a single parent household in which there is no male parent present, and we would therefore expect this to cause lower consumption among female led households.

We can also see that the average family size for men is slightly larger, with the proportion being as follows:

$$\frac{male\ family\ size}{female\ family\ size} = \frac{7.720943}{6.955882} \approx 1.109$$

We hypothesise that this can be potentially explained by the increased earning potential of males within Ugandan society, and therefore male led households have the ability to support larger households.

The average age of household heads is higher for female led households, with the difference as a proportion being as follows:

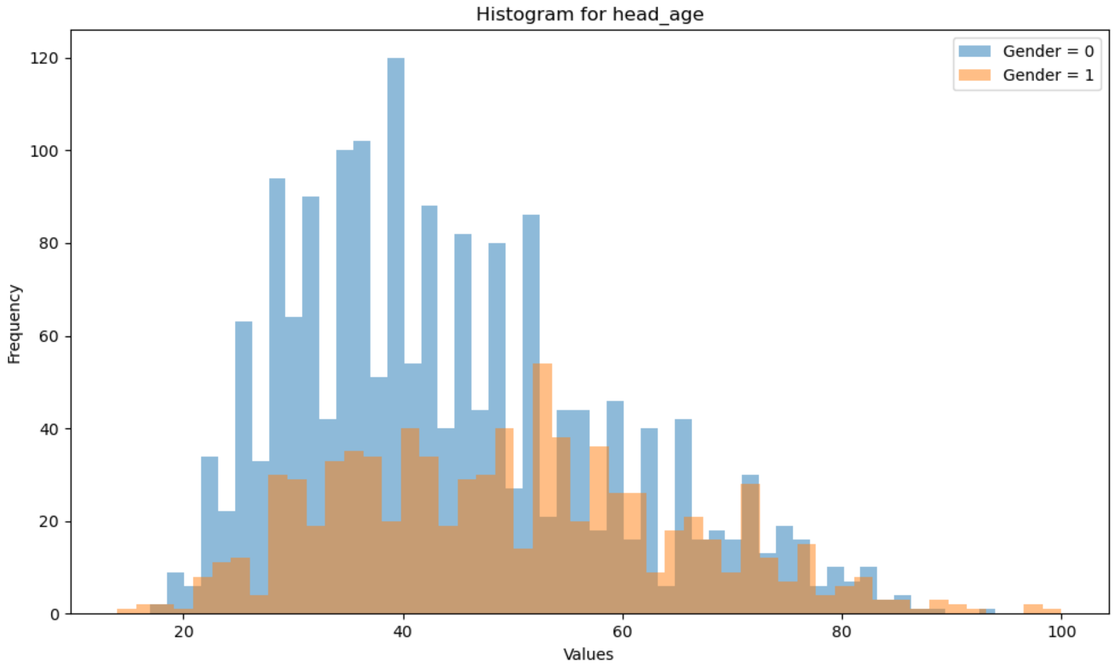
$$\frac{male\ head\ age}{female\ head\ age} = \frac{44.388546}{49.734069} \approx 0.893$$

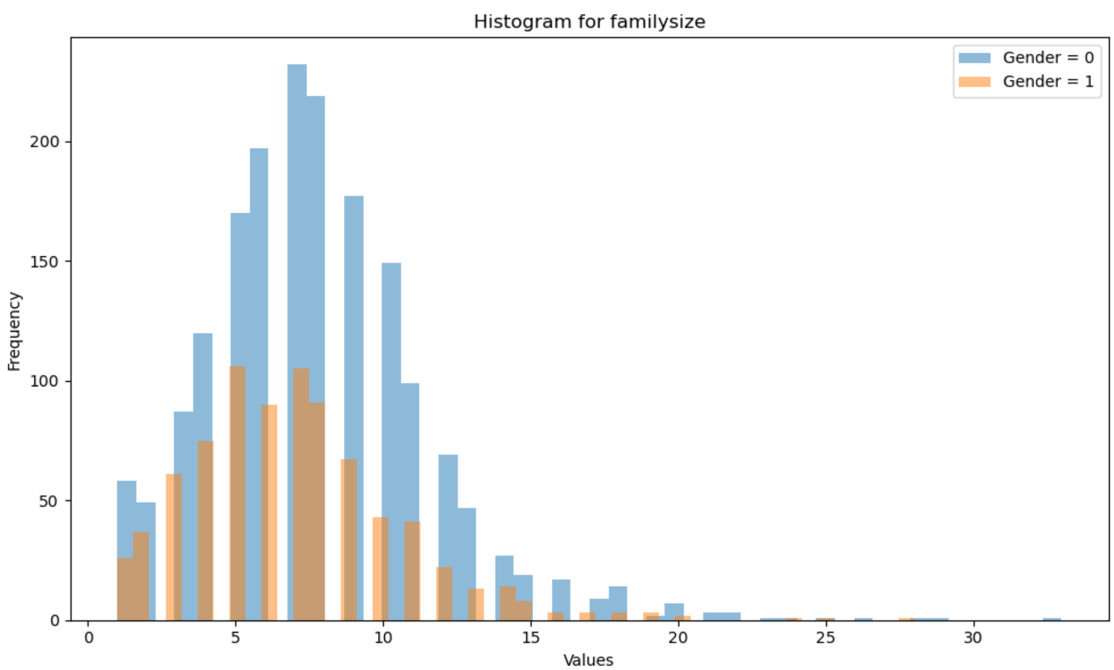
The absolute difference is years.

$$|male\ head\ age - female\ head\ age| = |44.388546 - 49.734069| \approx 5.346$$

We postulate that this is due to the fact that many female lead households will be female led because the previous male household head is deceased.

[70]:
```python
'''We have looked at the difference in values, but we can also
visualise the difference in distributions of values for male vs female led␣
↪households'''

columns_of_note_2 = ['head_age', 'familysize', 'female_dummy']

# Create subplots for each variable
fig, axes = plt.subplots(nrows=2, ncols=1, figsize=(10, 12))

# Plot histograms for each variable with male and female distributions on the␣
↪same line
for idx, column in enumerate(columns_of_note_2[:-1]):  # Exclude 'female_dummy'␣
↪from the loop
    for gender_value in df['female_dummy'].unique():
        axes[idx].hist(df[df['female_dummy'] == gender_value][column], bins=50,␣
↪alpha=0.5, label=f'Gender = {gender_value}')

    axes[idx].set_title(f'Histogram for {column}')
    axes[idx].set_xlabel('Values')
    axes[idx].set_ylabel('Frequency')
    axes[idx].legend(loc='upper right')

plt.tight_layout()
plt.show()
```





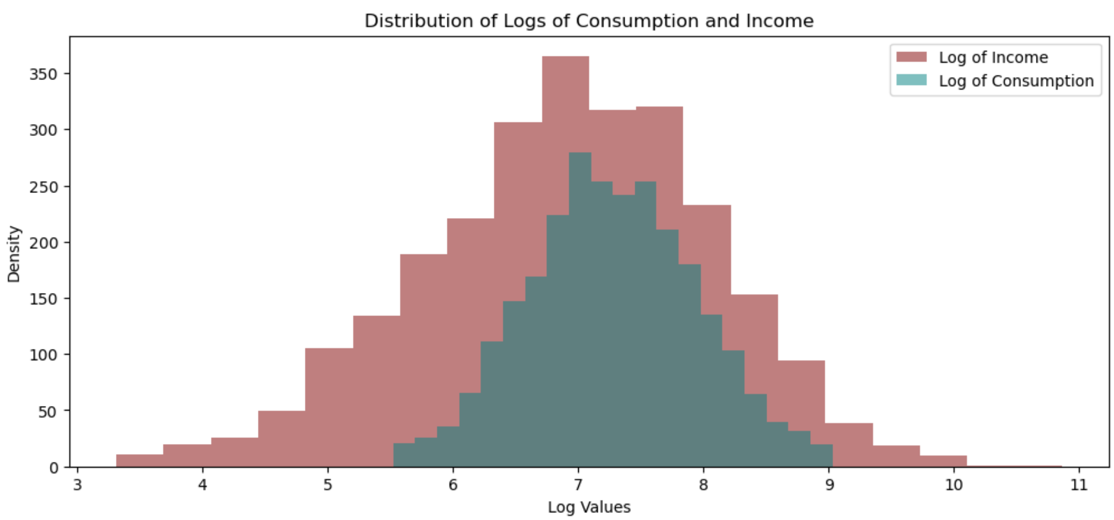
1.2 Exercise 2

1.2.1 Exercise 2a:

Create the variables log_c, log_inc, log_w that are the log of consumption, income, and wealth, respectively. Plot in the same graph the distribution of the log of consumption and the log of income. Do the distributions resemble some known distribution?

Is inequality higher in consumption or in income?

[71]:
```
df['log_c'] = np.log(df.consumption)
df['log_inc'] = np.log(df.income)
# Wealth has values of 0, so we have to account for them. If we make this too␣
↪small, it gives us an odd distribution and a very large variance in the logs␣
↪required for part b:
df['log_w'] = np.log(df.wealth + 1)

plt.figure(figsize=(12, 5))

plt.hist(df['log_inc'], bins=20, alpha=0.5, color ='maroon', label='Log of␣
↪Income')
plt.hist(df['log_c'], bins=20, alpha=0.5, color = 'teal', label='Log of␣
↪Consumption')

plt.xlabel('Log Values')
plt.ylabel('Density')
plt.title('Distribution of Logs of Consumption and Income')

plt.legend()
plt.show()
```



They resemble something close to a Gaussian, or normal, distribution.

Inequality is greater in terms of income.

1.2.2 Exercise 2b:

A commonly used statistic to measure inequality is the variance of the logs. Compute the variance of the log of consumption, of the log of income, and of the log of wealth. How do these measures of inequality in Uganda compare to the same measures of

inequality in the United States? Use table 3, column 5–PSID in De Magalhães, L., & Santaeulàlia-Llopis, R. (2018) for the comparison.

[72]:
```
var_income = df.log_inc.var()
var_consumption = df.log_c.var()
var_wealth = df.log_w.var()

# Print the variances for Uganda and the United States
print("Variance of logs of income:", round(var_income, 2), "(0.97)")

print("Variance of the logs of consumption:", round(var_consumption, 2), "(0.
↪79)")

print("Variance of the logs of wealth:", round(var_wealth, 2), "(2.11)")
```

```
Variance of logs of income: 1.34 (0.97)
Variance of the logs of consumption: 0.45 (0.79)
Variance of the logs of wealth: 2.98 (2.11)
```

The variance of the logs of income, consumption, and wealth in the De Magalhães, L., & Santaeulàlia-Llopis, R. (2018) are given in brackets

From this, we know that the UNPS data for uganda has a greater variance in terms of both income and wealth, but a lower variance in the log of the consumption.

1.2.3 Exercise 2c:

Measuring between rural and urban inequality in Uganda. Compute the average consump- tion, income, and wealth for rural and urban areas separately (groupby). Are the differences between the two areas large?

[73]:
```
average_inc = df.groupby('urban')['income'].mean()
average_c = df.groupby('urban')['consumption'].mean()
average_w = df.groupby('urban')['wealth'].mean()

# Print the average values and the differences
print("Average income:")
print("Rural:",round(average_inc.loc[0], 2),"USh")
print("Urban:",round(average_inc.loc[1], 2),"USh")

print("\nAverage consumption:")
print("Rural:",round(average_c.loc[0], 2),"USh")
print("Urban:",round(average_c.loc[1], 2),"USh")

print("\nAverage wealth:")
print("Rural:",round(average_w.loc[0], 2),"USh")
print("Urban:",round(average_w.loc[1], 2),"USh")
```

```
Average income:
Rural: 1713.96 USh
Urban: 2480.62 USh
```

```
Average consumption:
Rural: 1629.45 USh
Urban: 2544.22 USh

Average wealth:
Rural: 4521.0 USh
Urban: 6576.26 USh
```

1.2.4 Exercise 2d:

Measuring within rural and urban inequality in Uganda. Compute the variance of the log of consumption, income, and wealth for rural and urban areas separately.

[74]:
```
var_log_inc = df.groupby('urban')['log_inc'].var()
var_log_c = df.groupby('urban')['log_c'].var()
var_log_w = df.groupby('urban')['log_w'].var()

# Print the average values and the differences
print("Average income differences between urban and rural areas (log):")
print("Rural:",round(var_log_inc.loc[0], 2))
print("Urban:",round(var_log_inc.loc[1], 2))

print("\nAverage consumption differences between urban and rural areas (log):")
print("Rural:",round(var_log_c.loc[0], 2))
print("Urban:",round(var_log_c.loc[1], 2))

print("\nAverage wealth differences between urban and rural areas (log):")
print("Rural:",round(var_log_w.loc[0], 2))
print("Urban:",round(var_log_w.loc[1], 2))
```

```
Average income differences between urban and rural areas (log):
Rural: 1.34
Urban: 1.11

Average consumption differences between urban and rural areas (log):
Rural: 0.41
Urban: 0.44

Average wealth differences between urban and rural areas (log):
Rural: 2.52
Urban: 4.88
```

1.2.5 Exercise 2e:

Compute the Gini coefficient in consumption, in income, and in wealth in Uganda. Com- pare these values with the Gini coefficients in the United States—table 3, column 5–PSID in De Magalhães, L., & Santaeulàlia-Llopis, R. (2018)

[75]:
```
def gini(x):
    total = 0
    for i, xi in enumerate(x[:-1], 1):
        total = total + np.sum(np.abs(xi - x[i:]))
    return total / (len(x)**2 * np.mean(x))

# Print the gini values
print("Gini coefficient for income in Uganda (US):", round(gini(df.income), 2),␣
 ↪"(0.44)")

print("\nGini coefficient for consumption in Uganda (US):", round(gini(df.
 ↪consumption), 2), "(0.41)")

print("\nGini coefficient for wealth in Uganda (US):", round(gini(df.wealth),␣
 ↪2), "(0.79)")
```

```
Gini coefficient for income in Uganda (US): 0.56 (0.44)

Gini coefficient for consumption in Uganda (US): 0.37 (0.41)

Gini coefficient for wealth in Uganda (US): 0.66 (0.79)
```

1.2.6 Exercise 2f:

Compute the share of the wealth that the bottom 50 percent hold. Compute the share of the wealth that the top 10, 5, and 1 percent hold.

[76]:
```
# Calculate the total wealth
total_wealth = np.sum(df.wealth)

percentile_50 = np.percentile(df.wealth, 50)
percentile_90 = np.percentile(df.wealth, 90)
percentile_95 = np.percentile(df.wealth, 95)
percentile_99 = np.percentile(df.wealth, 99)

# Calculate the total wealth held by different percentiles
bottom_50_percent = np.sum(df[df.wealth <= percentile_50]['wealth'])
top_10_percent = np.sum(df[df.wealth >= percentile_90]['wealth'])
top_5_percent = np.sum(df[df.wealth >= percentile_95]['wealth'])
top_1_percent = np.sum(df[df.wealth >= percentile_99]['wealth'])

# Calculate the share of total wealth for different percentiles
bottom_50_percent_share = bottom_50_percent / total_wealth
top_10_percent_share = top_10_percent / total_wealth
top_5_percent_share = top_5_percent / total_wealth
top_1_percent_share = top_1_percent / total_wealth

# Print the results
```

```python
print("Share of the wealth that the bottom 50 percent hold:",␣
↪round(bottom_50_percent_share*100, 2), "%")
print("Share of the wealth that the top 10 percent hold:",␣
↪round(top_10_percent_share*100, 2), "%")
print("Share of the wealth that the top 5 percent hold:",␣
↪round(top_5_percent_share*100, 2), "%")
print("Share of the wealth that the top 1 percent hold:",␣
↪round(top_1_percent_share*100, 2),"%")
```

```
Share of the wealth that the bottom 50 percent hold: 8.23 %
Share of the wealth that the top 10 percent hold: 51.11 %
Share of the wealth that the top 5 percent hold: 35.25 %
Share of the wealth that the top 1 percent hold: 11.91 %
```

1.2.7 Exercise 2g:

Although in the last years, there has been a big debate on inequality, the debate has mostly focused on rich countries. From your results of this exercise, discuss whether inequality is relatively large in Uganda with respect to rich countries. From this study, we see that wealth inequality is comparitavily lower in Uganda than it is in the US and other richer nations. It should be noted though that the top 1% in Uganda will not have the same wealth levels of the top 1% in richer nations. It is only on a national level that we see Uganda has lower wealth disparities between its residents.

TL;DR: Rhe wealth inequality in Uganda is relatively small with respect with rish countries, when looking within-country, and not internationally.

1.2.8 Exercise 2h:

The few previous studies on income inequality in Africa had to rely on consumption measures to estimate income inequality. See, for example, Alvaredo & Gasparini (2005). Debate on the advantages and disadvantages of using consumption measures to study income inequality.

[ ]:
```python

```

1.3 Exercise 3

1.3.1 Exercise 3a:

Plot the lifecycle of the log of consumption for households where the head is male and for households where the head is female.

[82]:
```python
# Set the range as requested
df_filtered = df[(df['head_age'] >= 18) & (df['head_age'] <= 80)].copy() #␣
↪copy() means it doesnt replace the old dataframe, but rather makes a copy to␣
↪be edited as required

# Turn from 1 & 2 to Male and Female for ease
df_filtered['head_gender'] = df_filtered['head_gender'].map({1: 'Male', 2:␣
↪'Female'})
```

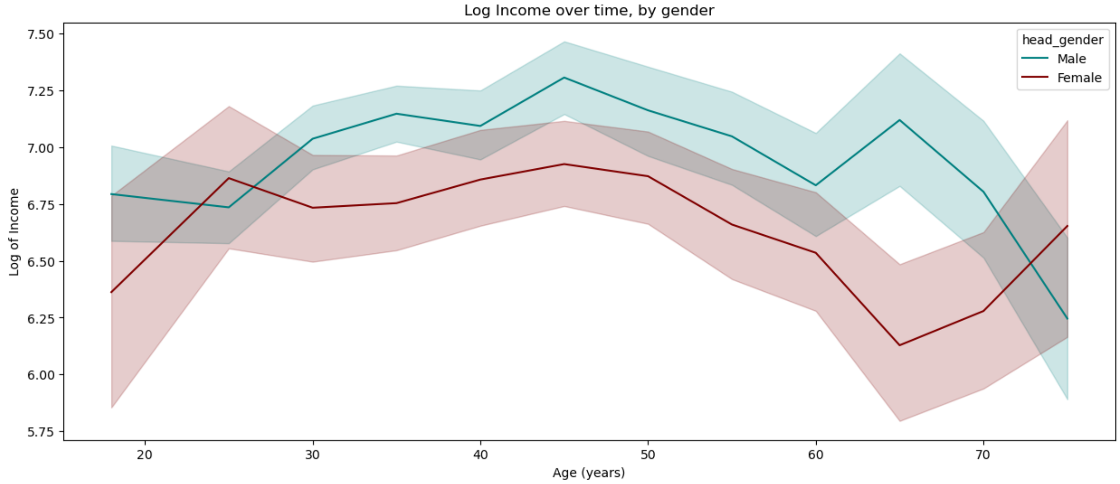
```python
# Create age bins (see lecture 2 notes)
age_bins = [18, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75, 80]
age_labels = [18, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75]
df_filtered['age_group'] = pd.cut(df_filtered['head_age'], bins=age_bins,␣
↪labels=age_labels)

# Give each gender a colour
gender_colors = {'Female': 'maroon', 'Male': 'teal'}

# Function to plot lifecycle by gender for a given financial metric
def plot_lifecycle(df, metric, title):
    plt.figure(figsize=(15, 6))
    sns.lineplot(data=df, x='age_group', y=metric, hue='head_gender',␣
↪palette=gender_colors, legend='full')
    plt.xlabel('Age (years)')
    plt.ylabel('Log of ' + title)
    plt.title("Log " + title + " over time, by gender")
    plt.show()
```

[78]:
```python
plot_lifecycle(df_filtered, 'log_c', 'Consumption')
```

1.3.2 Exercise 3b(i):

Redo the same plot but for the log of income

[79]:
```python
plot_lifecycle(df_filtered, 'log_inc', 'Income')
```



1.3.3 Exercise 3b(ii):

Redo the same plot but for the log of wealth

[83]:

```
  Cell In[83], line 1
    plot_lifecycle(df_filtered, 'log_w, 'Wealth')
                                               ^
SyntaxError: unterminated string literal (detected at line 1)
```

1.3.4 Exercise 3c:

What are the differences in the lifecycle of consumption, income, and wealth of households across the gender of the household heads? Comment your results.