Assignment 2

For this assignment you'll be looking at 2017 data on immunizations from the CDC. Your datafile for this assignment is in assets/NISPUF17.csv (assets/NISPUF17.csv). A data users guide for this, which you'll need to map the variables in the data to the questions being asked, is available at assets/NIS-PUF17-DUG.pdf (assets/NIS-PUF17-DUG.pdf). **Note: you may have to go to your Jupyter tree (click on the Coursera image) and navigate to the assignment 2 assets folder to see this PDF file).**

Question 1

Write a function called `proportion_of_education` which returns the proportion of children in the dataset who had a mother with the education levels equal to less than high school (<12), high school (12), more than high school but not a college graduate (>12) and college degree.

This function should return a dictionary in the form of (use the correct numbers, do not round numbers):

```
{"less than high school":0.2,
"high school":0.4,
"more than high school but not college":0.2,
"college":0.2}
```

In [1]:

```
'''
@author:  Steven Ponce
Date:     19 April 2021
'''

def proportion_of_education():

    # YOUR CODE HERE

    # Importing pandas library
    import pandas as pd

    # Loading the dataset
    df = pd.read_csv('assets/NISPUF17.csv')
    df = df[['EDUC1']]

    '''
    Input:
        EDUC1 - education of the mother
            one = Less than 12 years
            two = 12 years (high school)
            three = More than 12 years, not a college graduate
            four = College graduate

    Output:
        average_influenza_doses() returns the eturns the proportion of children in the dataset
        who had a mother with the education levels equal to less than high school (<12), high school (12),
        more than high school but not a college graduate (>12) and college degree
    '''
    one = len(df[df['EDUC1'] == 1])/len(df['EDUC1'])
    two = len(df[df['EDUC1'] == 2])/len(df['EDUC1'])
    three = len(df[df['EDUC1'] == 3])/len(df['EDUC1'])
    four = len(df[df['EDUC1'] == 4])/len(df['EDUC1'])

    return {'less than high school':one,
            'high school':two,
            'more than high school but not college':three,
            'college':four}
```

In [2]:

```
proportion_of_education()
```

Out[2]:

```
{'less than high school': 0.10202002459160373,
 'high school': 0.172352011241876,
 'more than high school but not college': 0.24588090637625154,
 'college': 0.47974705779026877}
```

In [3]:

```
assert type(proportion_of_education())==type({}), "You must return a dictionary."
assert len(proportion_of_education()) == 4, "You have not returned a dictionary with four items in it."
assert "less than high school" in proportion_of_education().keys(), "You have not returned a dictionary with the correct keys."
assert "high school" in proportion_of_education().keys(), "You have not returned a dictionary with the correct keys."
assert "more than high school but not college" in proportion_of_education().keys(), "You have not returned a dictionary with the correct keys."
assert "college" in proportion_of_education().keys(), "You have not returned a dictionary with the correct keys."
```

Question 2

Let's explore the relationship between being fed breastmilk as a child and getting a seasonal influenza vaccine from a healthcare provider. Return a tuple of the average number of influenza vaccines for those children we know received breastmilk as a child and those who know did not.

This function should return a tuple in the form (use the correct numbers:

```
(2.5, 0.1)
```

In [4]:

```
'''
@author:  Steven Ponce
Date:     19 April 2021
'''

def average_influenza_doses():

    # YOUR CODE HERE

    # Importing pandas library
    import pandas as pd

    # Loading the dataset
    df = pd.read_csv('assets/NISPUF17.csv')
    df = df[['CBF_01', 'P_NUMFLU']]

    df.isna().sum()
    df = df[df['P_NUMFLU'].notna()]

    '''
    Input:
        CBF_01 - child ever fed breast milk
            1 = yes
            2 = no
            77 = don't know
            99 = missing value

        P_NUMFLU - total number of seasonal influenza doses

    Output:
        average_influenza_doses() returns the average number of influenza vaccines for those children we know received
        breastmilk as a child and those who know did not.
    '''

    # Breastfeeding
    breastfeeding = (df[df['CBF_01'] == 1]['P_NUMFLU'].sum()) / (len(df[df['CBF_01'] == 1]))

    # Not breastfeeding
    not_breastfeeding = (df[df['CBF_01'] == 2]['P_NUMFLU'].sum()) / (len(df[df['CBF_01'] == 2]))

    return breastfeeding, not_breastfeeding
```

In [5]:

```
average_influenza_doses()
```

Out[5]:

```
(1.8799187420058687, 1.5963945918878317)
```

In [6]:

```
assert len(average_influenza_doses())==2, "Return two values in a tuple, the first for yes and the second for no."
```

Question 3

It would be interesting to see if there is any evidence of a link between vaccine effectiveness and sex of the child. Calculate the ratio of the number of children who contracted chickenpox but were vaccinated against it (at least one varicella dose) versus those who were vaccinated but did not contract chicken pox. Return results by sex.

This function should return a dictionary in the form of (use the correct numbers):

```
{"male":0.2,
"female":0.4}
```

Note: To aid in verification, the `chickenpox_by_sex()['female']` value the autograder is looking for starts with the digits `0.0077`.

In [7]:

```python
'''
@author:  Steven Ponce
Date:     19 April 2021
'''

def chickenpox_by_sex():

    # YOUR CODE HERE

    # Importing pandas library
    import pandas as pd

    # Loading the dataset
    df = pd.read_csv('assets/NISPUF17.csv')
    data = df[df['P_NUMVRC'].ge(1) & df['HAD_CPOX'].le(2)].loc[:,['P_NUMVRC','HAD_CPOX','SEX']]

    male_infected = len(data[data['P_NUMVRC'].ge(1) & (data['HAD_CPOX'].eq(1)) & (data['SEX'].eq(1))])
    male_not_infected = len(data[data['P_NUMVRC'].ge(1) & (data['HAD_CPOX'].eq(2)) & (data['SEX'].eq(1))])

    female_infected = len(data[data['P_NUMVRC'].ge(1) & (data['HAD_CPOX'].eq(1)) & (data['SEX'].eq(2))])
    female_not_infected = len(data[data['P_NUMVRC'].ge(1) & (data['HAD_CPOX'].eq(2)) & (data['SEX'].eq(2))])

    '''
    Input:
        HAD_CPOX - did child ever have chicken pox
            1 = yes
            2 = no
            77 = don't know
            99 = missing value

        P_NUMVRC - total number of varicella doses

        SEX - sex of child
            1 = Male
            2 = Female

    Output:
        chickenpox_by_sex() returns the ratio of the number of children who contracted chickenpox but were vaccinated
        against it (at least one varicella dose) versus those who were vaccinated but did not contract chicken pox.
        Return results by sex.
    '''

    male = male_infected / male_not_infected
    female = female_infected / female_not_infected

    return {'male':male,
            'female':female}
```

In [8]:

```python
chickenpox_by_sex()
```

Out[8]:

```
{'male': 0.009675583380762664, 'female': 0.0077918259335489565}
```

In [9]:

```python
assert len(chickenpox_by_sex())==2, "Return a dictionary with two items, the first for males and the second for females."
```

Question 4

A correlation is a statistical relationship between two variables. If we wanted to know if vaccines work, we might look at the correlation between the use of the vaccine and whether it results in prevention of the infection or disease [1]. In this question, you are to see if there is a correlation between having had the chicken pox and the number of chickenpox vaccine doses given (varicella).

Some notes on interpreting the answer. The `had_chickenpox_column` is either `1` (for yes) or `2` (for no), and the `num_chickenpox_vaccine_column` is the number of doses a child has been given of the varicella vaccine. A positive correlation (e.g., `corr > 0`) means that an increase in `had_chickenpox_column` (which means more no's) would also increase the values of `num_chickenpox_vaccine_column` (which means more doses of vaccine). If there is a negative correlation (e.g., `corr < 0`), it indicates that having had chickenpox is related to an increase in the number of vaccine doses.

Also, `pval` is the probability that we observe a correlation between `had_chickenpox_column` and `num_chickenpox_vaccine_column` which is greater than or equal to a particular value occurred by chance. A small `pval` means that the observed correlation is highly unlikely to occur by chance. In this case, `pval` should be very small (will end in `e-18` indicating a very small number).

[1] This isn't really the full picture, since we are not looking at when the dose was given. It's possible that children had chickenpox and then their parents went to get them the vaccine. Does this dataset have the data we would need to investigate the timing of the dose?

In [10]:

```python
def corr_chickenpox():

    # Importing libraries
    import scipy.stats as stats
    import numpy as np
    import pandas as pd

    # Loading the dataset
    df = pd.read_csv('assets/NISPUF17.csv')
    df = df[['P_NUMVRC', 'HAD_CPOX']]

    df = df[df['HAD_CPOX'].between(1,2)]
    df = df[df['P_NUMVRC'].notna()]

    '''
    Input:
        HAD_CPOX - did child ever have chicken pox
            1 = yes
            2 = no
            77 = don't know
            99 = missing value

        P_NUMVRC - total number of varicella doses

        scipy.stats.pearsonr(x, y)

            x(N,) - Input array.
            y(N,) - Input array.

    Output:
        corr_chickenpox() returns the Pearson correlation coefficient (corr) and two-tailed p-value (pval).
    '''

    # YOUR CODE HERE

    corr, pval = stats.pearsonr(df['HAD_CPOX'],df['P_NUMVRC'])

    return corr
```

In [11]:

```python
corr_chickenpox()
```

Out[11]:

```
0.07044873460147986
```

In [12]:

```python
assert -1<=corr_chickenpox()<=1, "You must return a float number between -1.0 and 1.0."
```