Inferential Analysis

Chi Square Test

With

Post Hoc Analysis

And

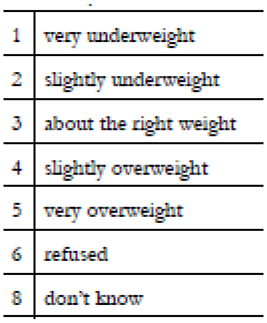
Python

AUTHOR:

LUIS ORELLANA ALTAMIRANO

Chi Square Test

“AddHealth” [1] was used through this post. The main idea is to check the relationship between two categorical variables, self-perception of weight (H1GH28) and sex (BIO_SEX). The “self-perception of weight’s levels are:



1	very underweight
2	slightly underweight
3	about the right weight
4	slightly overweight
5	very overweight
6	refused
8	don't know

After having got the outcome, the contingency table of observed counts is:

```
H1GH28    1    2     3     4    5  6  8
BIO_SEX
1        70  638  1736   641   55  2  5
2        58  297  1645  1167  183  1  5
```

And the chi square: 343.77945345342209

The p-value: 3.3401278903655964e-71

The chi square value is much higher than 3.84, and p-value is less than 0.003. It means these two variables have a strong relationship.

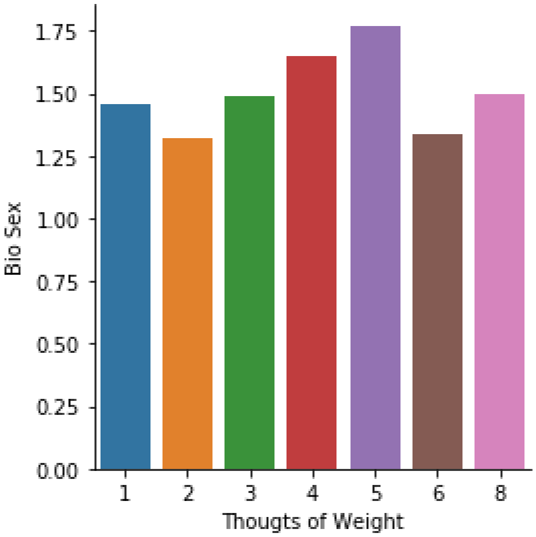
[1] Source: http://www.cpc.unc.edu/projects/addhealth

However, we don’t know which levels have the strongest relationship with the sex variable. Therefore, it’s necessary post hoc analysis.

The best result obtained is:

```
chi-square value: 264.811164
p value: 1.53E-59
COMP1v2  2.0   4.0
BIO_SEX
1        638   641
2        297  1167
COMP1v2       2.0       4.0
BIO_SEX
1        0.682353  0.354535
2        0.317647  0.645465
```

The relationship between sex value 2 (woman) and the values 2 to 4 of “self-perception of weight” is strong. This is depicted as follow:



Appendix

The program’s code is below:

```
import pandas
import numpy
import scipy.stats
import seaborn
import matplotlib.pyplot as plt

data = pandas.read_csv(‘addhealth_pds.csv’, low_memory=False)

data['BIO_SEX’]=data['BIO_SEX’].replace(0, numpy.nan)
data['H1GH28’]=data['H1GH28’].replace(0, numpy.nan)

data['BIO_SEX’] = pandas.to_numeric(data['BIO_SEX’], errors='coerce’)
data['H1GH28’] = pandas.to_numeric(data['H1GH28’], errors='coerce’)

#There is a six within the data
data = data[(data['BIO_SEX’] == 1) | (data['BIO_SEX’] == 2)]

ct=pandas.crosstab(data['BIO_SEX’], data['H1GH28’])
print (ct)

# column percentages
colsum=ct.sum(axis=0)
colpct=ct/colsum
print(colpct)

# chi-square
print ('chi-square value, p value, expected counts’)
cs= scipy.stats.chi2_contingency(ct)
print (cs)

# set variable types
data[“H1GH28”] = data[“H1GH28”].astype('category’)
# new code for setting variables to numeric:
data['BIO_SEX’] = pandas.to_numeric(data['BIO_SEX’], errors='coerce’)

# graph percent with nicotine dependence within each smoking frequency
group
seaborn.factorplot(x=“H1GH28”, y=“BIO_SEX”, data=data, kind=“bar”, ci=None)
plt.xlabel('Thougts of Weight’)
plt.ylabel('Bio Sex’)
```

```
###################Post Hoc Analysis

#make a copy of my new subsetted data
sub = data.copy()

for a in [1, 2, 3, 4, 5, 6]:
   for b in range((a + 1),9):

       if b != 7:

           print (’——————-’)
           recode1 = {a: a, b: b}
           sub['COMP1v2’]= data['H1GH28’].map(recode1)

           # contingency table of observed counts
           ct1=pandas.crosstab(sub['BIO_SEX’], sub['COMP1v2’])

           cs1= scipy.stats.chi2_contingency(ct1)
           if (cs1[0] > 3.84) & (cs1[1] < 0.003) :
               print ('chi-square value %f’%cs1[0])
               print ('p value %.2E’%cs1[1])

               print (ct1)

               # column percentages
               colsum=ct1.sum(axis=0)
               colpct=ct1/colsum
               print(colpct)
```