# Data Analysis Tools Week 2-ANOVA

April 10, 2023

[55]:
```
import pandas
import numpy
import seaborn
import matplotlib.pyplot as plt
import scipy.stats
import statsmodels.formula.api as smf
import statsmodels.stats.multicomp as multi
```

[56]:
```
data = pandas.read_csv('nesarc_pds.csv', low_memory=False)
```

[57]:
```
# Convert data types from 'Object' to 'Float'
data["S2AQ19"] = data["S2AQ19"].apply(pandas.to_numeric,errors="coerce")
data["S4AQ1"] = data["S4AQ1"].apply(pandas.to_numeric,errors="coerce")
data["S4AQ6A"] = data["S4AQ6A"].apply(pandas.to_numeric, errors="coerce")
data["S5Q1"] = data["S5Q1"].apply(pandas.to_numeric,errors="coerce")
data["S5Q3"] = data["S5Q3"].apply(pandas.to_numeric,errors="coerce")
data["S5Q8B"] = data["S5Q8B"].apply(pandas.to_numeric,errors="coerce")
```

[58]:
```
data['S2AQ19'].dtype
data['S4AQ1'].dtype
data['S4AQ6A'].dtype
data['S5Q1'].dtype
data['S5Q3'].dtype
data['S5Q8B'].dtype
```

[58]:
```
dtype('float64')
```

[59]:
```
# Reduce data set to drinkers <21yrs old
sub1=data[(data['S2AQ19']<=21)]
print (len(sub1))
```

```
4657
```

[60]:
```
sub2=sub1.copy()
```

[61]:
```
# Data Management Action 1: Set aside missing data
sub2['S2AQ19'] = sub2['S2AQ19'].replace(9,numpy.nan)
sub2['S4AQ1']=sub2['S4AQ1'].replace(9,numpy.nan)
```

```
sub2['S4AQ6A'] = sub2['S4AQ6A'].replace(9,numpy.nan)
sub2['S5Q1']=sub2['S5Q1'].replace(9,numpy.nan)
sub2['S5Q3']=sub2['S5Q3'].replace(9,numpy.nan)
sub2["S5Q8B"] = sub2["S5Q8B"].replace(9,numpy.nan)
```

[62]:
```
# Data Management Action 2: Create secondary varialbe 'MentalHealthScore'
sub2['MentalHealthScore']=sub1['S4AQ1']+sub1['S5Q1']+sub1['S5Q3']
```

[63]:
```
# Convert new variable to numeric
sub2["MentalHealthScore"] = sub2["MentalHealthScore"].apply(pandas.
 ↪to_numeric,errors="coerce")
```

[64]:
```
# Data Management Action 3: Grouping values within individual variables to␣
 ↪create MentalHealthCondition based off MentalHealthScore
def BiPolarIndicator (row):
    if row['MentalHealthScore'] == 3:
        return 'Yes'
    if row['MentalHealthScore'] > 3:
        return 'No'

sub2['BiPolarIndicator'] = sub2.apply (lambda row: BiPolarIndicator (row),␣
 ↪axis=1)

print('Top 25 Rows Confirming BiPolarIndicator Calculation')
sub3=sub2[['IDNUM', 'S2AQ19','S4AQ1','S5Q1', 'S5Q3', 'MentalHealthScore',␣
 ↪'BiPolarIndicator']]
sub3.head(25)
```

```
Top 25 Rows Confirming BiPolarIndicator Calculation
```

[64]:

| | IDNUM | S2AQ19 | S4AQ1 | S5Q1 | S5Q3 | MentalHealthScore | BiPolarIndicator |
|---|---|---|---|---|---|---|---|
| 1 | 2 | 21.0 | 2.0 | 2.0 | 2.0 | 6.0 | No |
| 3 | 4 | 16.0 | 2.0 | 1.0 | 2.0 | 5.0 | No |
| 4 | 5 | 18.0 | 2.0 | 2.0 | 2.0 | 6.0 | No |
| 5 | 6 | 18.0 | 2.0 | 2.0 | 2.0 | 6.0 | No |
| 6 | 7 | 18.0 | 1.0 | 1.0 | 1.0 | 3.0 | Yes |
| 8 | 9 | 21.0 | 1.0 | 2.0 | 1.0 | 4.0 | No |
| 9 | 10 | 17.0 | 2.0 | 1.0 | 2.0 | 5.0 | No |
| 12 | 13 | 21.0 | 1.0 | 2.0 | 2.0 | 5.0 | No |
| 16 | 17 | 18.0 | 2.0 | 2.0 | 2.0 | 6.0 | No |
| 17 | 18 | 20.0 | 1.0 | 2.0 | 2.0 | 5.0 | No |
| 19 | 20 | 18.0 | 1.0 | 2.0 | 1.0 | 4.0 | No |
| 21 | 22 | 19.0 | 2.0 | 2.0 | 2.0 | 6.0 | No |
| 24 | 25 | 21.0 | 2.0 | 2.0 | 2.0 | 6.0 | No |
| 30 | 31 | 19.0 | 2.0 | 2.0 | 2.0 | 6.0 | No |
| 31 | 32 | 17.0 | 1.0 | 2.0 | 1.0 | 4.0 | No |
| 37 | 38 | 20.0 | 1.0 | 2.0 | 2.0 | 5.0 | No |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 39 | 40 | 20.0 | 2.0 | 2.0 | 2.0 | 6.0 | No |
| 40 | 41 | 16.0 | 1.0 | 2.0 | 2.0 | 5.0 | No |
| 41 | 42 | 18.0 | 2.0 | 2.0 | 2.0 | 6.0 | No |
| 44 | 45 | 15.0 | 1.0 | 2.0 | 1.0 | 4.0 | No |
| 45 | 46 | 19.0 | 2.0 | 2.0 | 2.0 | 6.0 | No |
| 51 | 52 | 20.0 | 2.0 | 2.0 | 2.0 | 6.0 | No |
| 52 | 53 | 19.0 | 2.0 | 2.0 | 2.0 | 6.0 | No |
| 53 | 54 | 20.0 | 2.0 | 2.0 | 2.0 | 6.0 | No |
| 54 | 55 | 19.0 | 2.0 | 2.0 | 2.0 | 6.0 | No |

[65]:
```
#sub3['AgeGroup']=pandas.cut(sub3.S2AQ19,[5, 12, 18, 21])
#sub3.head(50)
```

[66]:
```
#categorical quantitative variable based on customized splits with cut function
#splits age into 3 groups (5 - 12, 13 - 17, 18 - 21) - list start and end at␣
 ↪first and then end point of the others
## need to filter this down to bipolar = yes to keep it simple; otherwise a␣
 ↪multilevel categorical analysis is possible
sub3['AgeGroup'] = pandas.cut(sub3.S2AQ19, [4, 12, 18, 21])

print (pandas.crosstab(sub3['S2AQ19'], sub3['AgeGroup']))
```

```
AgeGroup  (4, 12]  (12, 18]  (18, 21]
S2AQ19
5.0           17         0         0
6.0            1         0         0
7.0            1         0         0
8.0            2         0         0
10.0           7         0         0
11.0           1         0         0
12.0          11         0         0
13.0           0        15         0
14.0           0        41         0
15.0           0       113         0
16.0           0       258         0
17.0           0       414         0
18.0           0      1112         0
19.0           0         0       640
20.0           0         0       742
21.0           0         0      1280
```

[67]:
```
# contingency table of observed counts - depression
ct1=pandas.crosstab(sub3['S4AQ1'], sub3['AgeGroup'])
print(ct1)
```

```
AgeGroup  (4, 12]  (12, 18]  (18, 21]
S4AQ1
```

```
1.0            10       645       844
2.0            30      1282      1779
```

[68]:
```
# contingency table of observed counts - elation
ct2=pandas.crosstab(sub3['S5Q1'], sub3['AgeGroup'])
print(ct2)
```

```
AgeGroup  (4, 12]  (12, 18]  (18, 21]
S5Q1
1.0             3       199       206
2.0            37      1716      2413
```

[69]:
```
# contingency table of observed counts - irritability
ct3=pandas.crosstab(sub3['S5Q3'], sub3['AgeGroup'])
print(ct3)
```

```
AgeGroup  (4, 12]  (12, 18]  (18, 21]
S5Q3
1.0             4       265       282
2.0            36      1657      2339
```

[70]:
```
# contingency table of observed counts - bipolar indicator
ct4=pandas.crosstab(sub3['BiPolarIndicator'], sub3['AgeGroup'])
print(ct4)
```

```
AgeGroup          (4, 12]  (12, 18]  (18, 21]
BiPolarIndicator
No                     39      1873      2585
Yes                     1        80        77
```

[71]:
```
# column percentages
colsum=ct4.sum(axis=0)
colpct=ct4/colsum
print(colpct)
```

```
AgeGroup          (4, 12]  (12, 18]  (18, 21]
BiPolarIndicator
No                  0.975  0.959037  0.971074
Yes                 0.025  0.040963  0.028926
```

[72]:
```
# chi-square
print('Chi-square value, p value, expected counts')
cs4=scipy.stats.chi2_contingency(ct4)
print(cs4)
```

```
Chi-square value, p value, expected counts
(5.076154523350214, 0.07901818511506745, 2, array([[3.86423201e+01,
1.88671128e+03, 2.57164640e+03],
```

```
       [1.35767991e+00, 6.62887218e+01, 9.03535983e+01]]))
```

[ ]:
```
#Interpretaion: The Chi-square test shows that the categorical analysis shows␣
↪the null hypothesis is accepted;
#there is no relationship between age when a person starts drinking and the␣
↪presence of bipolar mental illness
```

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]: