In [ ]:
```
#                                          Session 18 Assignment
#*******************************************************************************************
#                                          Problem Statement 1
#
#*******************************************************************************************
# Problem Statement 1:

#Is gender independent of education level? A random sample of 395 people were surveyed and each person was asked to repor
#       High School    Bachelors   Masters   Ph.d.   Total
# Female 60             54          46        41      201
# Male   40             44          53        57      194
# Total  100            98          99        98      395
# Question:  Are gender and education level dependent at 5% level of significance?
# In other words, given the data collected above, is there a relationship between the gender of an individual and the lev
#*******************************************************************************************
# its an exmaple of Chi square stats as probability of a category independent of another category is asked to find out

import numpy as np
import pandas as pd

# Let the null hypothesis and alternate hyothesis be
'''
Ho : Gender and Education are independent of each other
Ha : Gender and Education are dependent on each other

'''
# WE will be using chi square statistics as we have to show gender and education category dependence
# O is observed frequency as given in the table and Expected frequency which we have to calculate
# Degrees of Freedom here will be product of both the category Degrees of freedom
GenCount = 2
EduCount = 4
Df = (GenCount-1)*(EduCount-1)
print("The Degrees of Freedom  both categories is : %d " % Df)
print('-'*80)
# k2 = Sum(O - E)**2/E

# Putting the Data in DataFrame
df1 = pd.DataFrame({'HighSchool':[60,40,100],'Bachelors':[54,94,98],'Masters':[46,53,99],'Phd':[41,57,98],'Total':[201,19
print(" DAtaFrame having Observed values for Gender wise Education levels ")
print('-'*80)
print(df1)
```

```
# Expected Frequency for combination of each category is product of total of the each category
# Expected Frequency(Gender,Education) = GenTot *EduTot/TotalPeople
EfH = round((201*100)/395,2)
EfB = round((201*98)/395,2)
EfM = round((201*99)/395,2)
EfP = round((201*98)/395,2)
EmH = round((194*100)/395,2)
EmB = round((194*98)/395,2)
EmM = round((194*99)/395,2)
EmP = round((194*98)/395,2)

# Expected Frequency Data FRame
df2 = pd.DataFrame({'HighSchool':[EfH,EmH,(EfH+EmH)],'Bachelors':[EfB,EmB,(EfB+EmB)],'Masters':[EfM,EmM,(EfM+EmM)],'Phd':
print("\nDataFrame having Expected frequencies for gender wise Education levels")
print('-'*80)
print(df2)

# To find the chi square stats, finding the sum of squares of Diffe in Observed and Expected Frequeny
ChiSquare = round(((df1['HighSchool'][0]-df2['HighSchool'][0])**2/df2['HighSchool'][0]),2)+ round(((df1['Bachelors'][0]-d
ChiSquare = ChiSquare + round(((df1['Masters'][0]-df2['Masters'][0])**2/df2['Masters'][0]),2)+ round(((df1['Phd'][0]-df2[

ChiSquare = ChiSquare + round(((df1['HighSchool'][1]-df2['HighSchool'][1])**2/df2['HighSchool'][1]),2)+ round(((df1['Bach
ChiSquare = ChiSquare + round(((df1['Masters'][1]-df2['Masters'][1])**2/df2['Masters'][1]),2)+ round(((df1['Phd'][1]-df2[

print('-'*80)
print(" The Chi Square statistics obtained are : %.2f" %ChiSquare)
print(" Refering the Chi table for getting the 5% significance and Df of 3, Chi critical :  7.815")
print('-'*80)
print(" We observe that Chi sq stats > than Chi critical and hence , we reject the null hypothesis in favour of Alternate
print(" There fore Gender and Education are not independent , there is some dependency between them ")
```

In [ ]:
```
#*****************************************************************************************************
#                                          Problem Statement 2
#
#*****************************************************************************************************
#Using the following data, perform a oneway analysis of variance using α=.05. Write up the results in APA format.
#[Group1: 51, 45, 33, 45, 67] [Group2: 23, 43, 23, 43, 45] [Group3: 56, 76, 74, 87, 56]

# importing packages and setting an alias
import numpy as np
import pandas as pd

# creating a DataFrame with the given groups to form each column
dict = {'Group1' : [51,45,33,45,67],'Group2':[23,43,23,43,45],'Group3':[56,76,74,87,56]}


df1 = pd.DataFrame(dict)


#print(df1)

''' let the null hypothesis be there is no difference in the means of the groups and alternative hypothesis they are not
Ho : mu1 = mu2 = mu3
H1 : mu1 != mu2 != mu3

'''
# For one way analysis of variance , that means we need to do the F test of variances
# finding the mean of the groups
mu1 = np.mean(df1.Group1)
mu2 = np.mean(df1.Group2)
mu3 = np.mean(df1.Group3)

# Total mean of the groups is  as same number  elemts in each group
muTot = (mu1 +mu2 +mu3)/3
print("The means of the GRoups are : %.2f , %.2f , %.2f"%(mu1,mu2,mu3))
print("Total Number of Elements in all groups n :%d "% (len(df1.Group1)+len(df1.Group2)+len(df1.Group3)) )
print("Total number of groups are  k : 3")
k = 3
n = (len(df1.Group1)+len(df1.Group2)+len(df1.Group3))

print("The Total mean of all groups is : %.2f" %muTot)
```

```
# getting the Variance for each group as differnt columns in the DataFrame
df1['SumofSquare(Gp1-mu1)'] = (df1['Group1']-mu1)**2
df1['SumofSquare(Gp2-mu2)'] = (df1['Group2']-mu2)**2
df1['SumofSquare(Gp3-mu3)'] = (df1['Group3']-mu3)**2

print('-'*80)
print(" The Data Frame containing the Group Data \n",'-'*80)
print(df1)
# number of elements in each group
l = len(df1['Group1'])

# F test = (Sumof squares(Difference of Group and Total mean)/k-1)/(sumofsquares(variance with respective group mean)/n-k
# Degree of freedome numerator is dfn = k-1  . and of denominator is dfd = n-k
dfn = k-1
dfd = n-k

#Numerator is  Sumof squares(Difference of Group and Total mean)/k-1
DiffTotmean = (((mu1-muTot)**2 + (mu2-muTot)**2 + (mu3-muTot)**2)*l)/dfn

print(" The Vairances with respect to Total mean : %.2f"%DiffTotmean)
#print(DiffTotmean)

#Denominator is  Sum of variances of all the group elements /
DiffGpmean = (sum(df1['SumofSquare(Gp1-mu1)']) + sum(df1['SumofSquare(Gp2-mu2)']) + sum(df1['SumofSquare(Gp3-mu3)']))/dfd

print(" The Vairances of elements with respect to Group mean : %.2f"%DiffGpmean)
#print(DiffGpmean)

# Fstatistics score is
Fscore  = DiffTotmean/DiffGpmean

# F critical Value from the F table with dfn and dfd at alpha=0.05
print(" The Numerator Degree of Freedom and Denominator : %d, %d "%(dfn,dfd))

Fcrit = 3.8853
print('-'*80)

print("The F critical value for these degrees and alpha 0.05 from F table is : %.2f"%Fcrit)
print("The Fstatistics score is : %.2f"%Fscore)

print(" AS F statistics > Fcrit , we reject the Null Hypothesis in favour of Alternate Hypthesis ")
print(" Therefore the means of the groups are all different ")
```

In [ ]:

```
#*****************************************************************************************************
#                                        Problem Statement 3
#
#*****************************************************************************************************
# Calculate F Test for given 10, 20, 30, 40, 50 and 5,10,15, 20, 25.

# For 10, 20, 30, 40, 50:

# F test is the test to check the variances relation between the population
# F =  sigma1**2/sigma2**2
# For 10,20,30,40,50
import numpy as np

# count and degrees of freedom for first sample of numbers
n1 = 5
Dfn = n1-1

X = np.array([10,20,30,40,50])
mu1 = round(np.mean(X),2)

# std of numpy gives by default population deviation , to get sample deviation setting ddof parameter
sd1 = np.std(X,ddof=1)
Var1 = sd1**2

print(" The Mean and Variance of the first array is : %.2f , %.2f "%(mu1,Var1))

# count and degrees of freedom for 2nd sample

n2 = 5
Dfd = n2-1

Y = np.array([5,10,15,20,25])
mu2 = round(np.mean(Y),2)

# std of numpy gives by default population deviation , to get sample deviation setting ddof parameter
sd2 = np.std(Y,ddof=1)
Var2 = sd2**2

print(" The Mean and Variance of the first array is : %.2f , %.2f "%(mu2,Var2))

print('-'*80)
```

```
# the F test is the variance ratio test , s1**2/s2**2

Fscore = Var1/Var2

print(" The F Test for [10,20,30,40,50] is : %.2f" %Fscore)
```