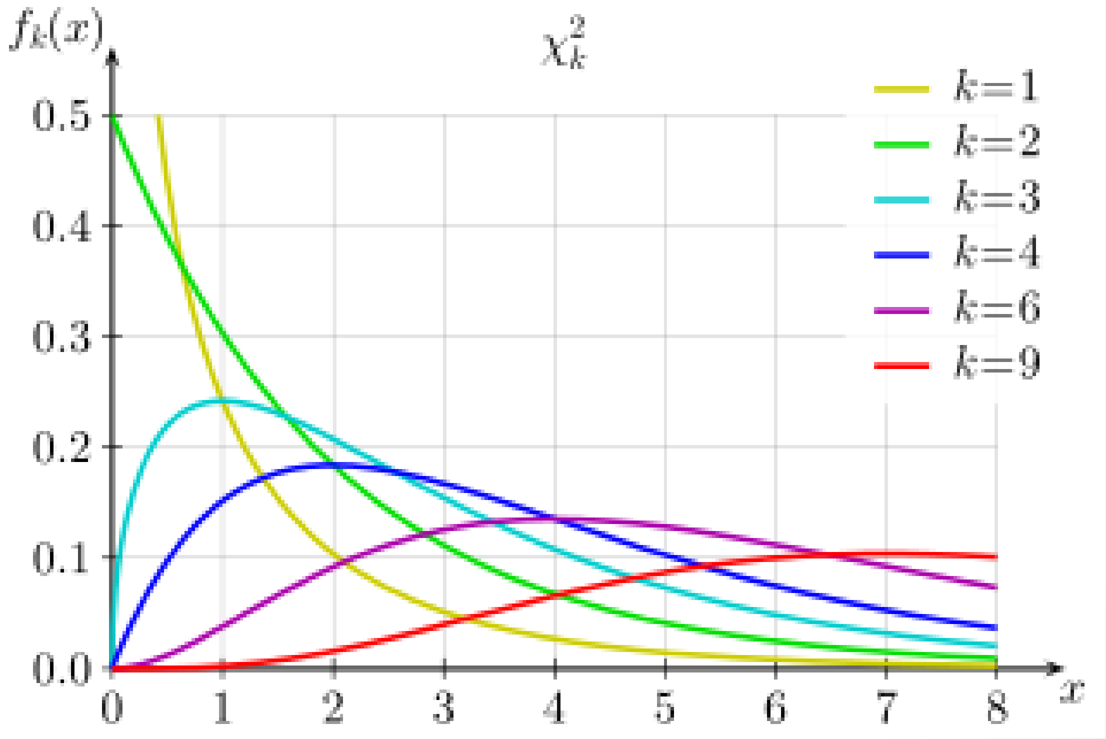
Chi-Squared Test

Chi-square Test

- There are two types of chi-square tests. Both use the chi-square statistic and distribution for different purposes

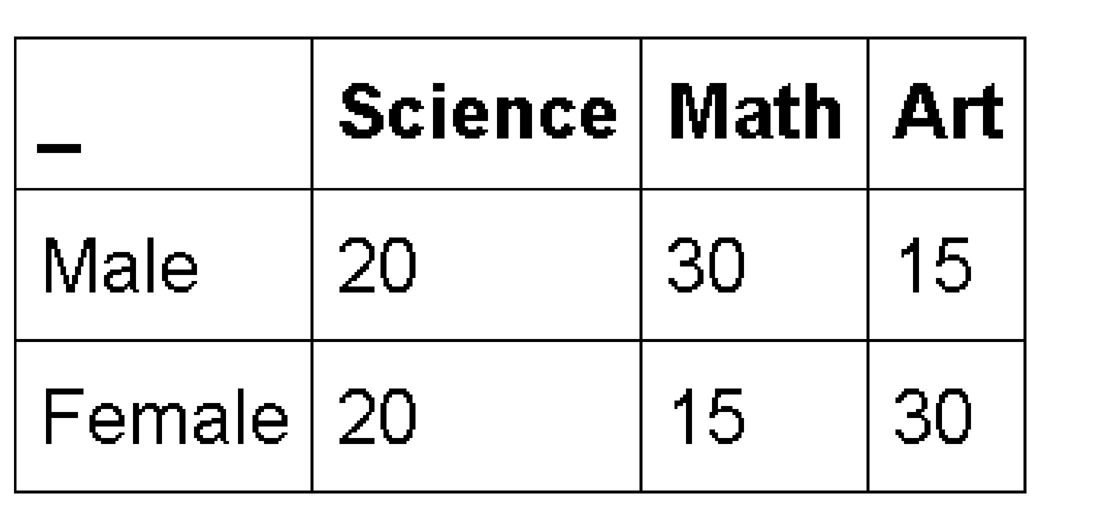


Types of Chi-Square Test

- Goodness of Fit
  - Determines if a sample data matches a population
- Test of Independance
  - Compares two variables in a contingency table to see if they are related.
  - Tests to see whether distributions of categorical variables differ from each another.

Problem Statement

Q. Is the Marks obtained by the students depended on Gender as per the given Data

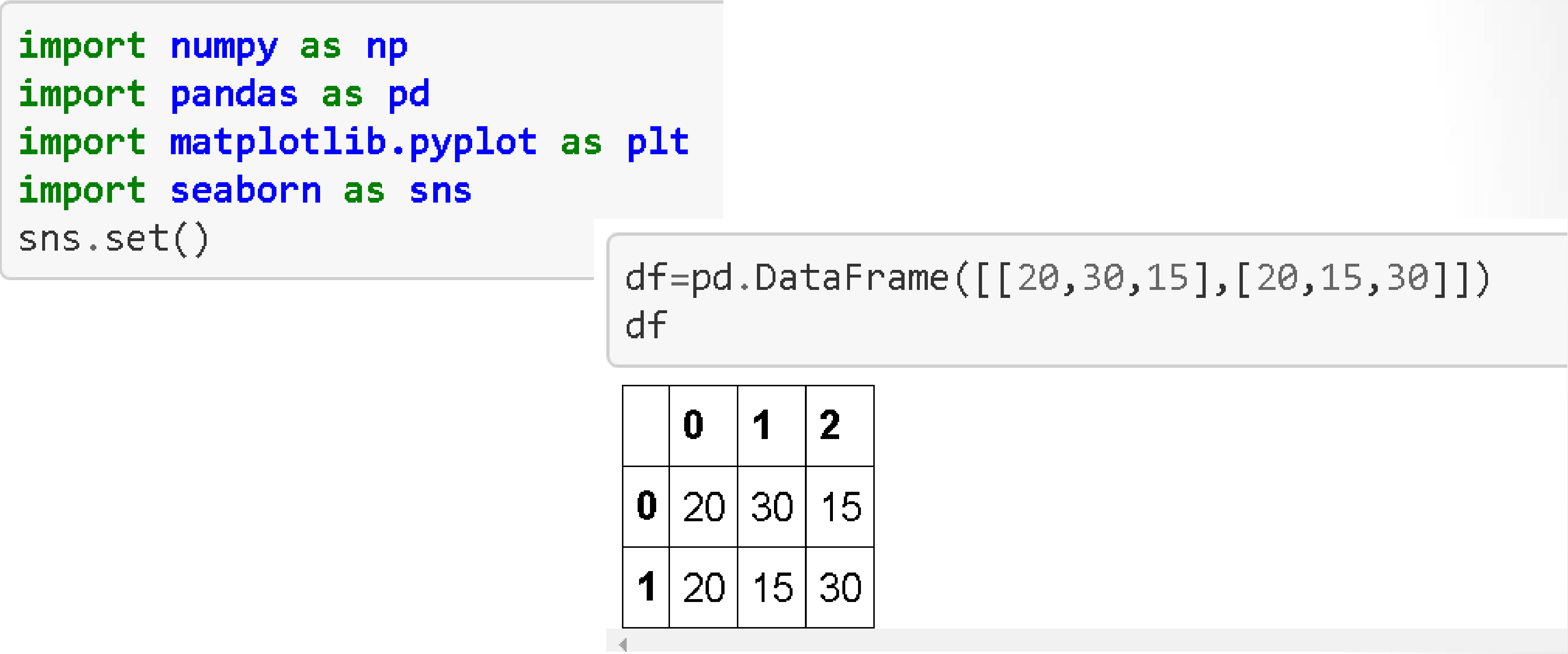


–	Science	Math	Art
Male	20	30	15
Female	20	15	30

Null Hypothesis and Alternate Hypothesis

- Null Hypothesis
  - Two Categorical Variables are independant
  - H0 : Marks independant of Gender
- AlternateHypothesis
  - Two Categorical Variables are dependant
  - H0 : Marks dependant of Gender

Import Libraries and Create a dataframe



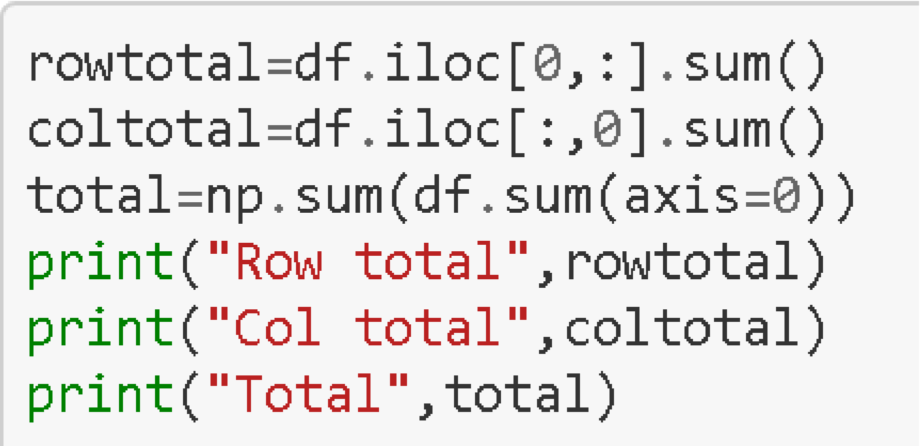
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
sns.set()
```

```
df=pd.DataFrame([[20,30,15],[20,15,30]])
df
```

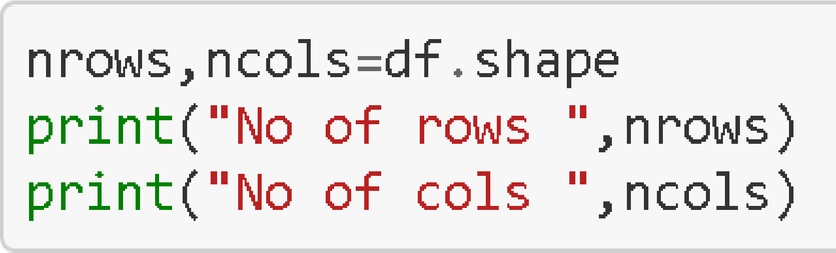
	0	1	2
0	20	30	15
1	20	15	30

Calculate the Expected Matrix

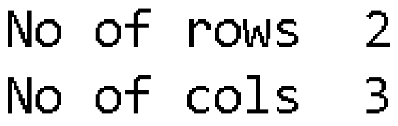
$$Expectedvalue = \frac{Row_{total} * column_{total}}{Total}$$



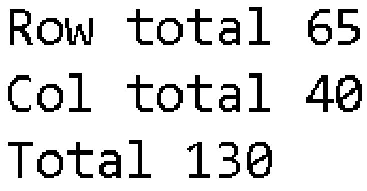
```
rowtotal=df.iloc[0,:].sum()
coltotal=df.iloc[:,0].sum()
total=np.sum(df.sum(axis=0))
print("Row total",rowtotal)
print("Col total",coltotal)
print("Total",total)
```



```
nrows,ncols=df.shape
print("No of rows ",nrows)
print("No of cols ",ncols)
```

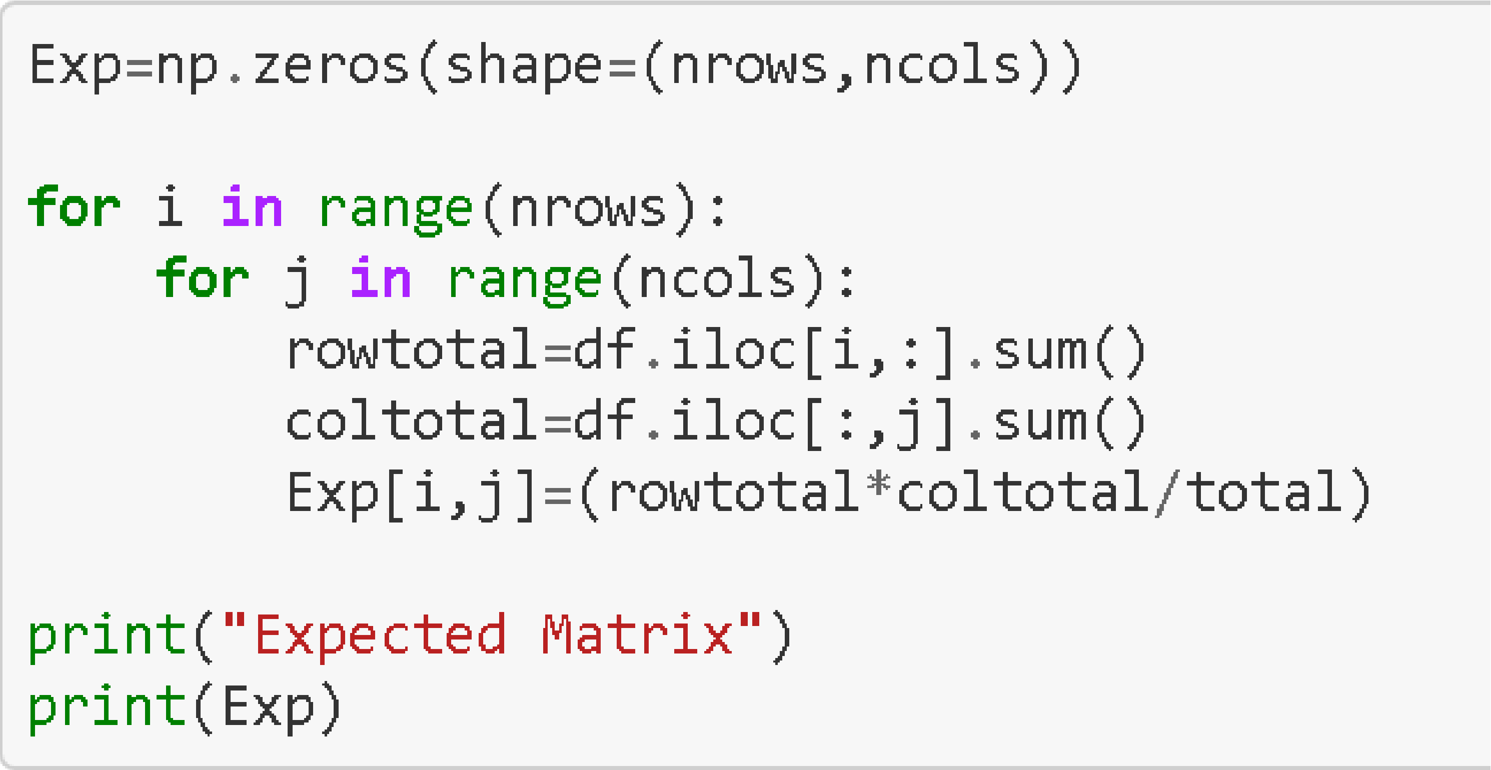


```
No of rows  2
No of cols  3
```

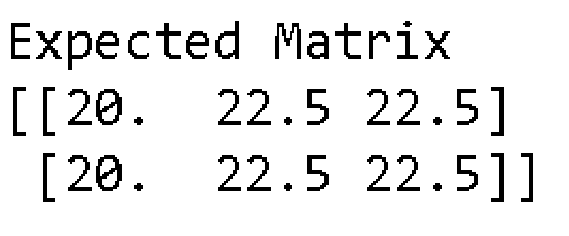


```
Row total 65
Col total 40
Total 130
```

Expected Matrix



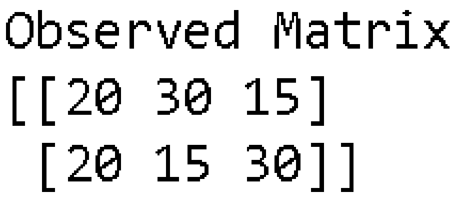
```
Exp=np.zeros(shape=(nrows,ncols))

for i in range(nrows):
    for j in range(ncols):
        rowtotal=df.iloc[i,:].sum()
        coltotal=df.iloc[:,j].sum()
        Exp[i,j]=(rowtotal*coltotal/total)

print("Expected Matrix")
print(Exp)
```



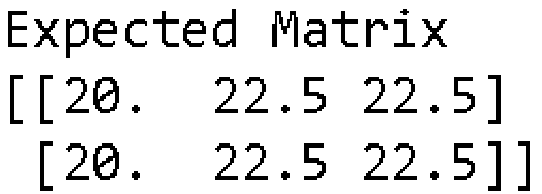
```
Expected Matrix
[[20.   22.5 22.5]
 [20.   22.5 22.5]]
```

Chi Statistics

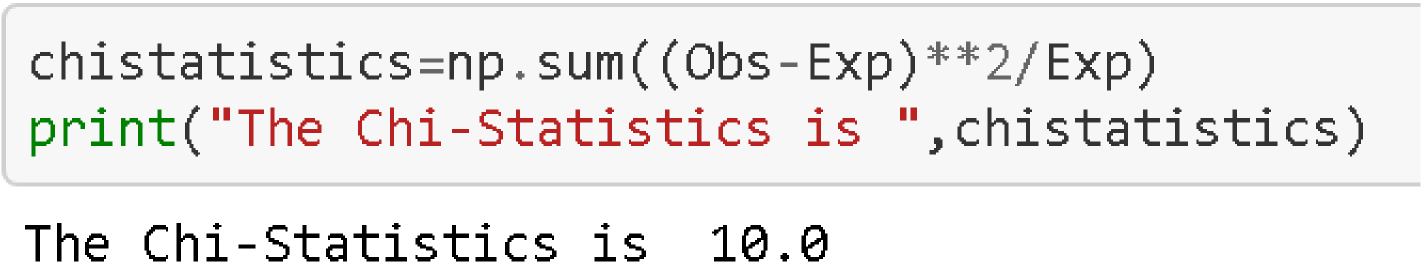
$$Chi - Statistics = \sum \frac{(Observed - Expected)^2}{Expected}$$



```
Observed Matrix
[[20 30 15]
 [20 15 30]]
```



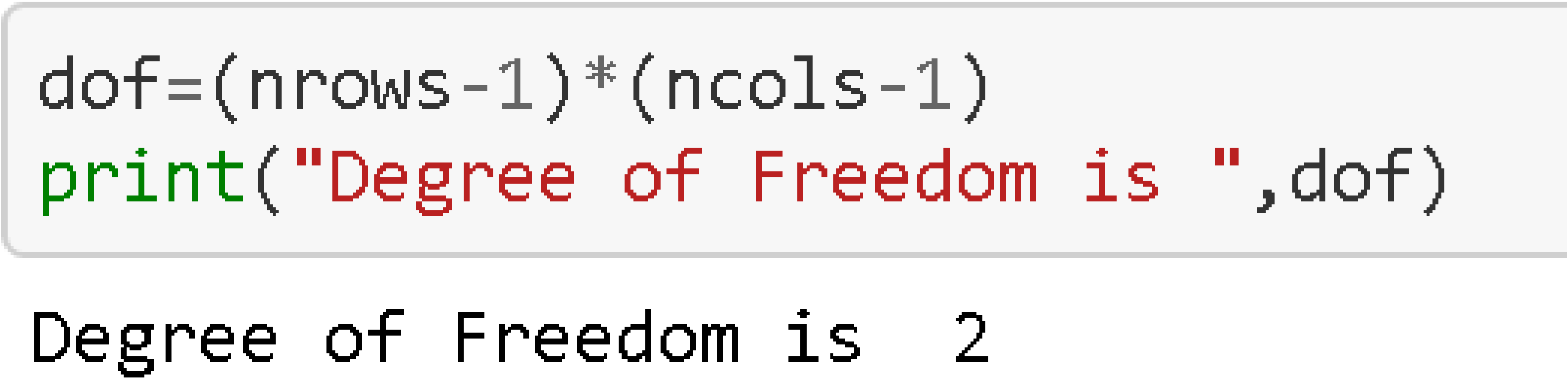
```
Expected Matrix
[[20.   22.5 22.5]
 [20.   22.5 22.5]]
```



```python
chistatistics=np.sum((Obs-Exp)**2/Exp)
print("The Chi-Statistics is ",chistatistics)
```

```
The Chi-Statistics is  10.0
```

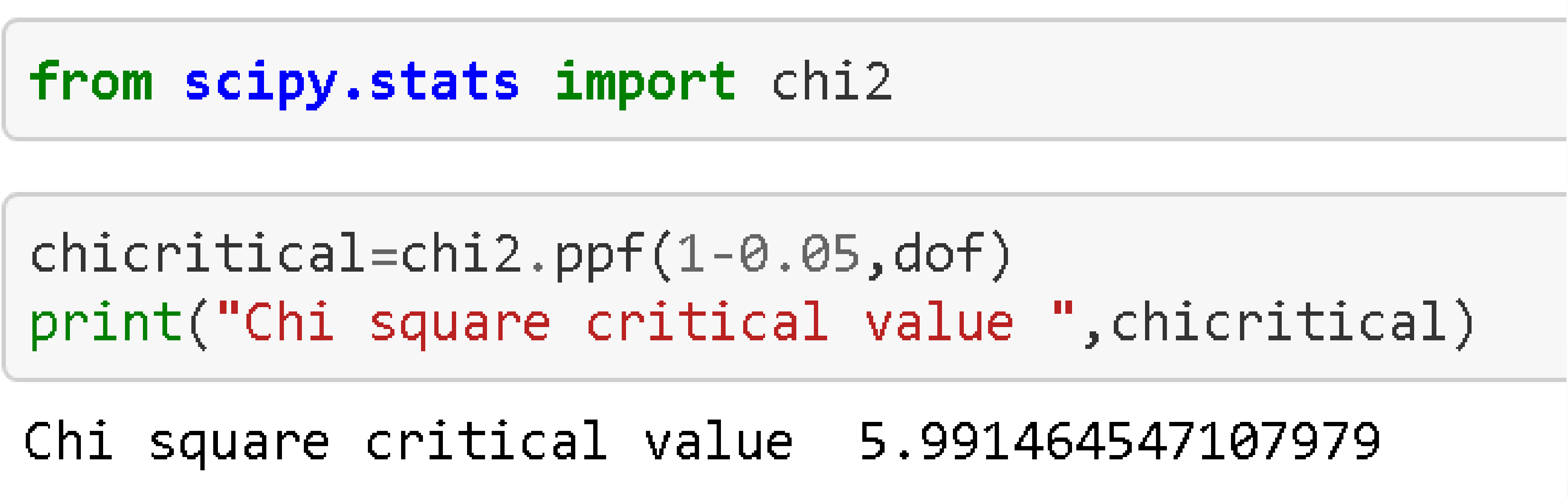
Degree of Freedom



```python
dof=(nrows-1)*(ncols-1)
print("Degree of Freedom is ",dof)
```

```
Degree of Freedom is  2
```

Critical Value

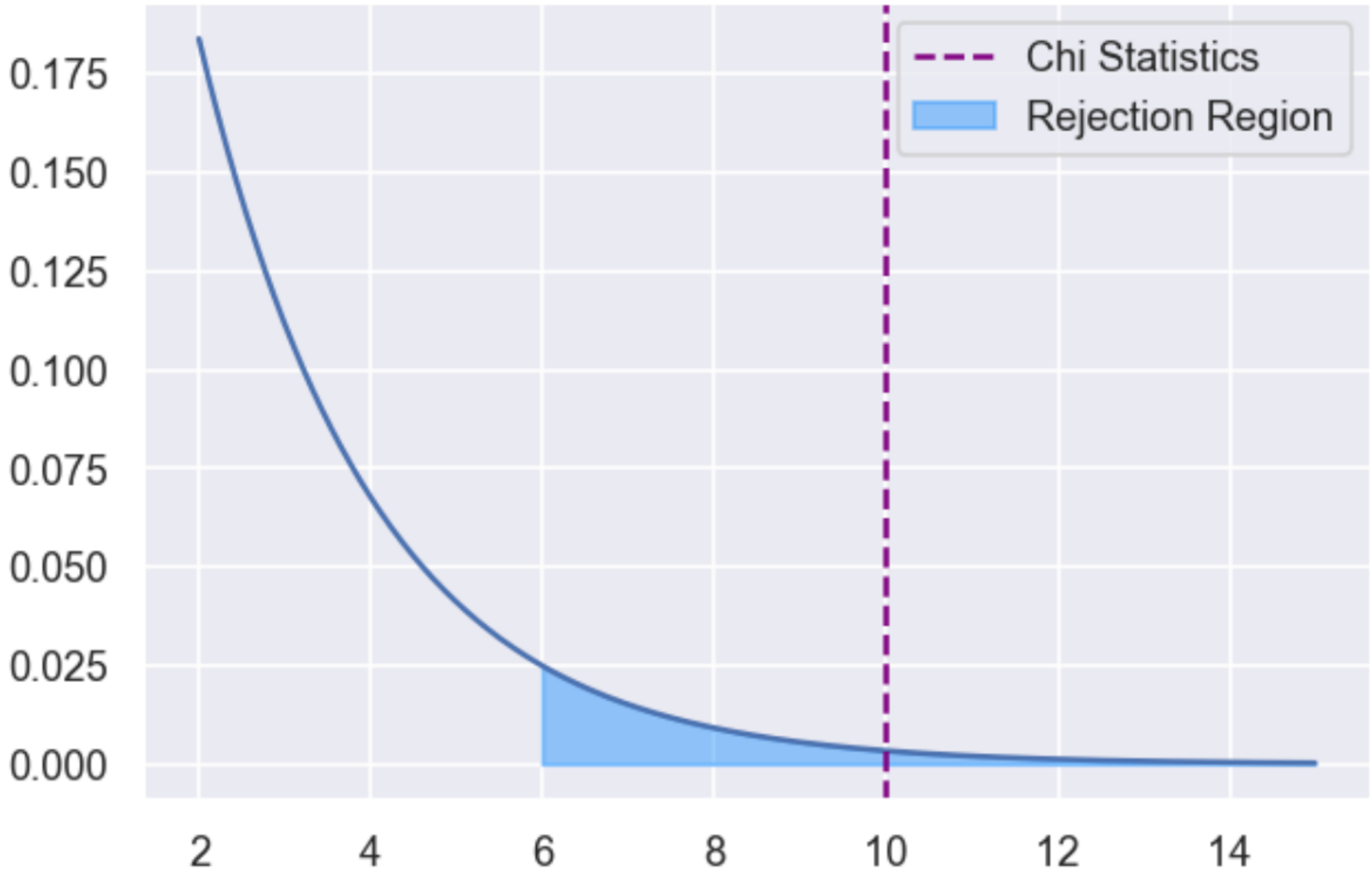


```
from scipy.stats import chi2
```

```
chicritical=chi2.ppf(1-0.05,dof)
print("Chi square critical value ",chicritical)
```

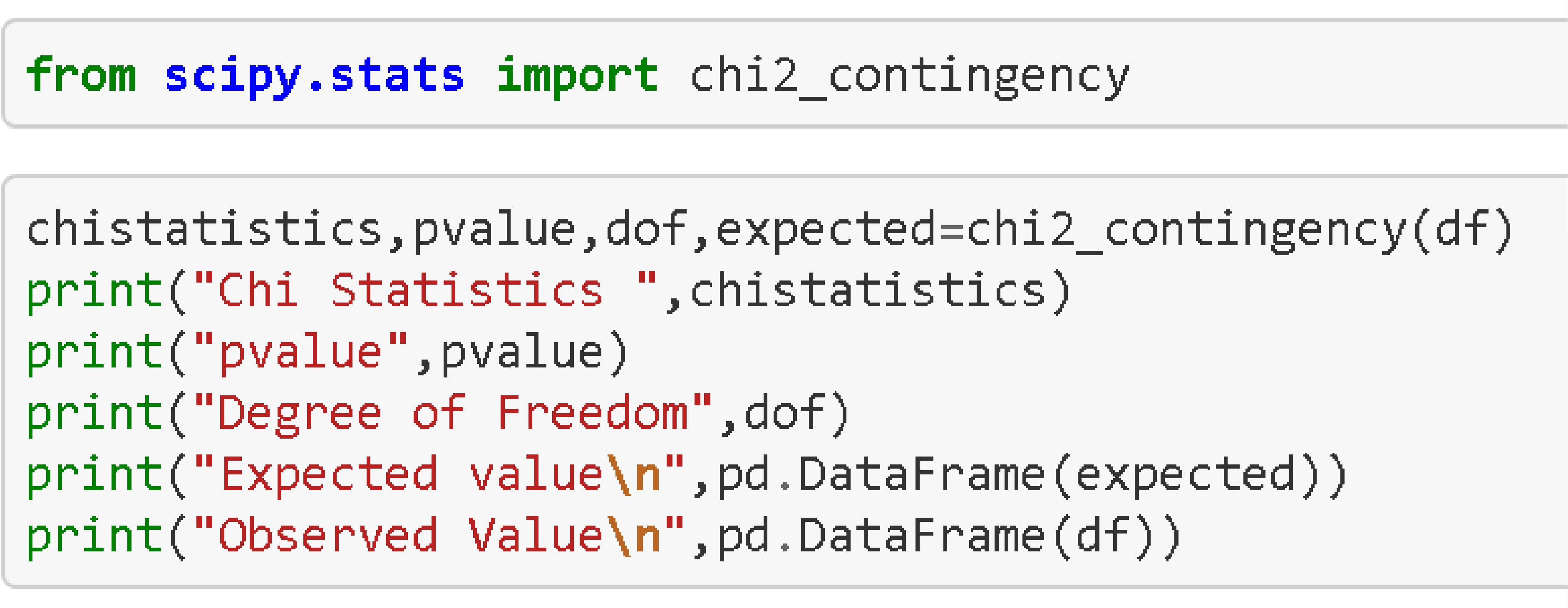
```
Chi square critical value  5.991464547107979
```

Plot the Findings



Chi-Square test using Inbuilt chi2_contingency in scipy

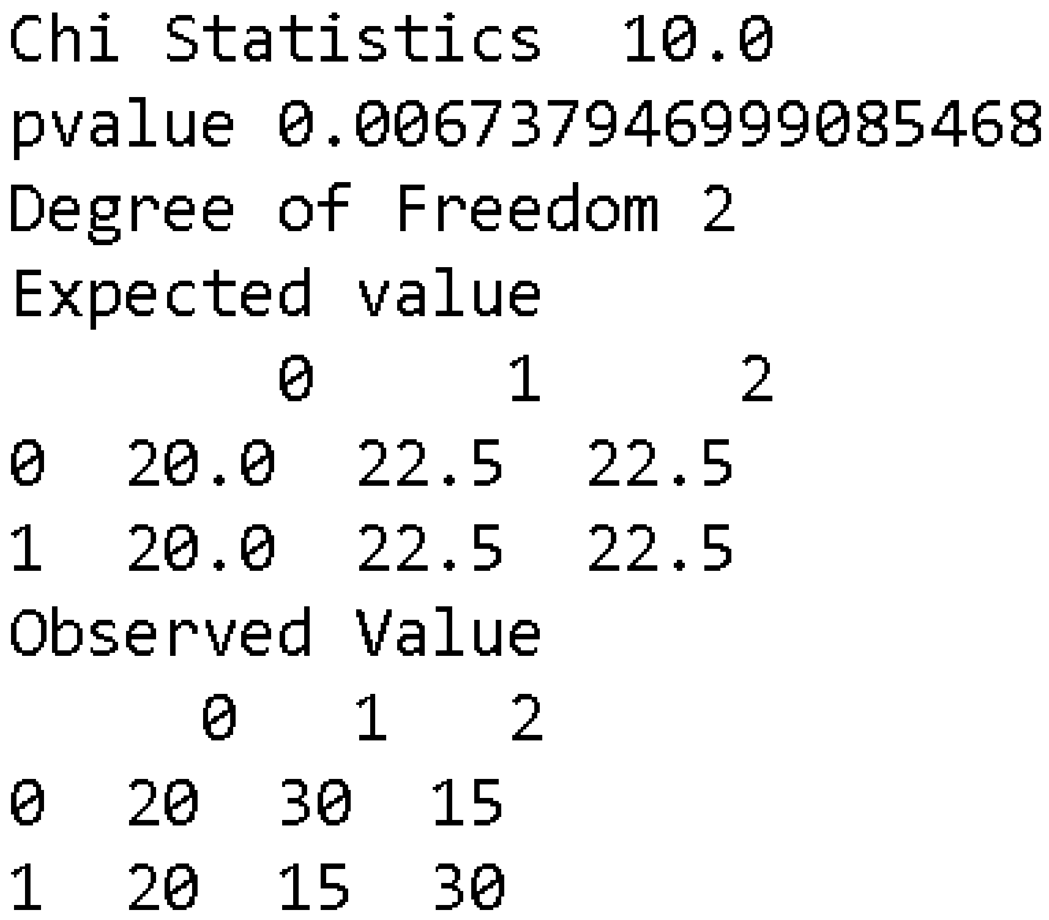
Chi-squared test using chi2_contingency method



```
from scipy.stats import chi2_contingency
```

```
chistatistics,pvalue,dof,expected=chi2_contingency(df)
print("Chi Statistics ",chistatistics)
print("pvalue",pvalue)
print("Degree of Freedom",dof)
print("Expected value\n",pd.DataFrame(expected))
print("Observed Value\n",pd.DataFrame(df))
```

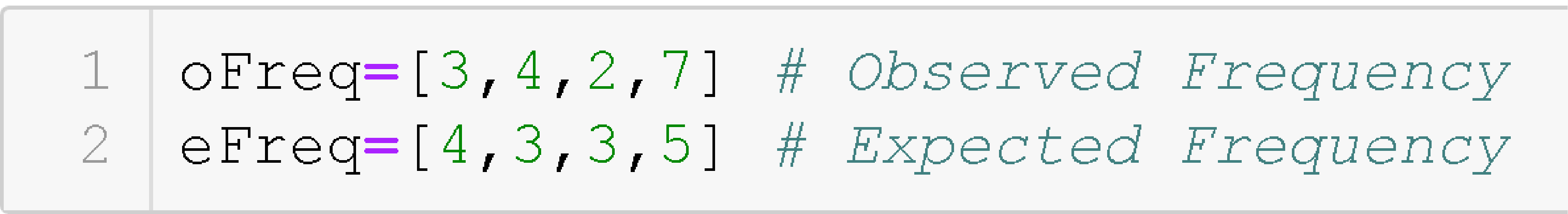
Output of the code



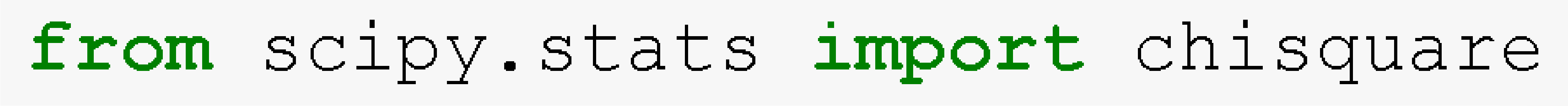
```
Chi Statistics  10.0
pvalue 0.006737946999085468
Degree of Freedom 2
Expected value
       0     1     2
0  20.0  22.5  22.5
1  20.0  22.5  22.5
Observed Value
     0   1   2
0  20  30  15
1  20  15  30
```

IF FREQUECIES ARE GIVEN

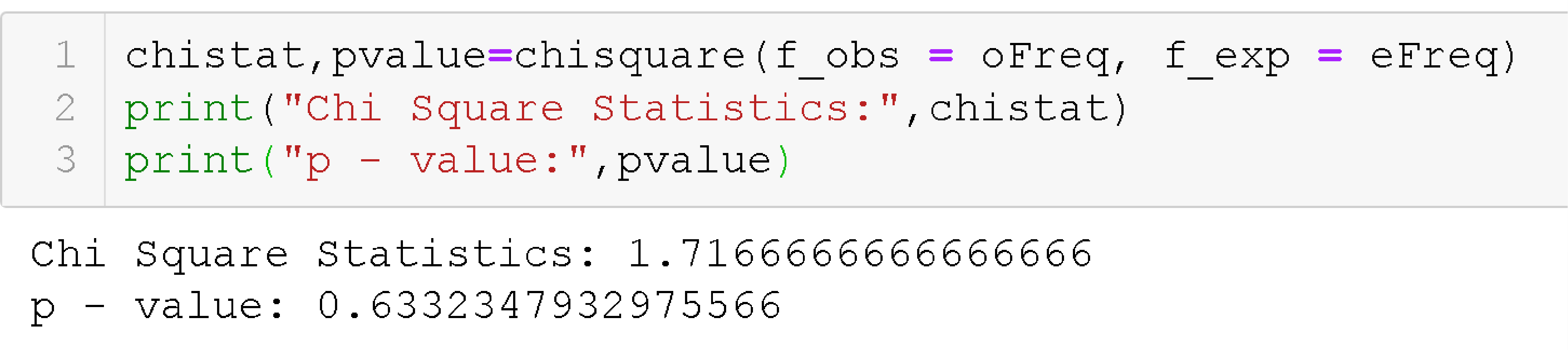
Consider the following scenario



```
oFreq=[3,4,2,7] # Observed Frequency
eFreq=[4,3,3,5] # Expected Frequency
```



```python
from scipy.stats import chisquare
```



```python
chistat,pvalue=chisquare(f_obs = oFreq, f_exp = eFreq)
print("Chi Square Statistics:",chistat)
print("p - value:",pvalue)
```

```
Chi Square Statistics: 1.7166666666666666
p - value: 0.6332347932975566
```