

# Installing packages

```
In [1]: pip install matplotlib pandas numpy statsmodels scipy sklearn nbconvert
```

Requirement already satisfied: matplotlib in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (3.5.1)

Requirement already satisfied: pandas in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (1.4.2)

Requirement already satisfied: numpy in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (1.22.3)

Requirement already satisfied: statsmodels in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (0.13.2)

Requirement already satisfied: scipy in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (1.8.0)

Requirement already satisfied: sklearn in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (0.0)

Requirement already satisfied: nbconvert in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (6.5.0)

Requirement already satisfied: kiwisolver<=1.0.1 in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (from matplotlib) (1.4.2)

Requirement already satisfied: cyclor<=0.10 in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (from matplotlib) (0.11.0)

Requirement already satisfied: pillow<=6.2.0 in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (from matplotlib) (9.1.0)

Requirement already satisfied: python-dateutil<=2.7 in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (from matplotlib) (2.8.2)

Requirement already satisfied: pyparsing<=2.2.1 in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (from matplotlib) (3.0.8)

Requirement already satisfied: packaging<=20.0 in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (from matplotlib) (21.3)

Requirement already satisfied: fonttools<=4.22.0 in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (from matplotlib) (4.32.0)

Requirement already satisfied: pytz<=2020.1 in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (from pandas) (2022.1)

Requirement already satisfied: patsy<=0.5.2 in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (from statsmodels) (0.5.2)

Requirement already satisfied: scikit-learn in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (from sklearn) (1.0.2)

Requirement already satisfied: beautifulsoup4 in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (from nbconvert) (4.11.1)

Requirement already satisfied: jupyter-core<=4.7 in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (from nbconvert) (4.9.2)

Requirement already satisfied: pygments<=2.4.1 in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (from nbconvert) (2.11.2)

Requirement already satisfied: tinycss2 in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (from nbconvert) (1.1.1)

Requirement already satisfied: traitlets<=5.0 in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (from nbconvert) (5.1.1)

Requirement already satisfied: nbformat<=5.1 in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (from nbconvert) (5.3.0)

Requirement already satisfied: nbclient<=0.5.0 in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (from nbconvert) (0.6.0)

Requirement already satisfied: mistune<2,>=0.8.1 in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (from nbconvert) (0.8.4)

Requirement already satisfied: entrypoints<=0.2.2 in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (from nbconvert) (0.4)

Requirement already satisfied: jupyterlab-pygments in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (from nbconvert) (0.2.1)

Requirement already satisfied: defusedxml in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (from nbconvert) (0.7.1)

Requirement already satisfied: bleach in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (from nbconvert) (5.0.0)

Requirement already satisfied: jinja2<=3.0 in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (from nbconvert) (3.1.1)

Requirement already satisfied: MarkupSafe>=2.0 in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (from nbconvert) (2.1.1)

Requirement already satisfied: pandocfilters>=1.4.1 in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (from nbconvert) (1.5.0)

Requirement already satisfied: jupyter-client>=6.1.5 in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (from nbclient>=0.5.0->nbconvert) (7.2.2)

Requirement already satisfied: nest-asyncio in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (from nbclient>=0.5.0->nbconvert) (1.5.5)

Requirement already satisfied: tornado>=6.0 in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (from jupyter-client>=6.1.5->nbclient>=0.5.0->nbconvert) (6.1)

Requirement already satisfied: pyzmq>=22.3 in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (from jupyter-client>=6.1.5->nbclient>=0.5.0->nbconvert) (22.3.0)

Requirement already satisfied: jsonschema>=2.6 in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (from nbformat>=5.1->nbconvert) (4.4.0)

Requirement already satisfied: fastjsonschema in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (from nbformat>=5.1->nbconvert) (2.15.3)

Requirement already satisfied: pyparsing!=0.17.0,!=0.17.1,!=0.17.2,>=0.14.0 in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (from jsonschema>=2.6->nbformat>=5.1->nbconvert) (0.18.1)

Requirement already satisfied: attrs>=17.4.0 in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (from jsonschema>=2.6->nbformat>=5.1->nbconvert) (21.4.0)

Requirement already satisfied: importlib-resources>=1.4.0 in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (from jsonschema>=2.6->nbformat>=5.1->nbconvert) (5.7.0)

Requirement already satisfied: zipp>=3.1.0 in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (from importlib-resources>=1.4.0->jsonschema>=2.6->nbformat>=5.1->nbconvert) (3.8.0)

Requirement already satisfied: six in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (from patsy>=0.5.2->statsmodels) (1.16.0)

Requirement already satisfied: soupsieve>1.2 in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (from beautifulsoup4->nbconvert) (2.3.2)

Requirement already satisfied: webencodings in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (from bleach->nbconvert) (0.5.1)

Requirement already satisfied: threadpoolctl>=2.0.0 in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (from scikit-learn->sklearn) (3.1.0)

Requirement already satisfied: joblib>=0.11 in /home/gorigan/miniconda3/envs/pesquisa-bio/lib/python3.8/site-packages (from scikit-learn->sklearn) (1.1.0)

Note: you may need to restart the kernel to use updated packages.

## Importing Libraries

```
In [2]: %matplotlib inline
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
from numpy.random import seed
from numpy.random import randn
from statsmodels.graphics.gofplots import qqplot
import numpy as np
```

```

from scipy import stats
from math import sqrt
from numpy.random import seed
from numpy.random import randn
from numpy import mean
from scipy.stats import sem
from scipy.stats import t

```

```

In [3]: # function for calculating the t-test for two independent samples
def independent_ttest(data1, data2, alpha):
    # calculate means
    mean1, mean2 = mean(data1), mean(data2)
    # calculate standard errors
    se1, se2 = sem(data1), sem(data2)
    # standard error on the difference between the samples
    sed = sqrt(se1**2.0 + se2**2.0)
    # calculate the t statistic
    t_stat = (mean1 - mean2) / sed
    # degrees of freedom
    df = len(data1) + len(data2) - 2
    # calculate the critical value
    cv = t.ppf(1.0 - alpha, df)
    # calculate the p-value
    p = (1.0 - t.cdf(abs(t_stat), df)) * 2.0
    # return everything
    return t_stat, df, cv, p

```

## Toy Samples

```

In [4]: mu2, sigma2 = 25, 5 # mean and standard deviation
s2 = np.random.normal(mu2, sigma2, 5000)
mu, sigma = 30, 5 # mean and standard deviation
s = np.random.normal(mu, sigma, 5000)
zz = [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]

```

## Population

Each distribution is a total of eggs laid during 10 days for each female individual

```

In [5]: F= [3,0,5,2,3,8,1,6,3,5,5,2,0,6,2,2,4,3,3,3]
FM= [3,5,2,1,0,1,4,8,4,4,2,2,4,3,2,2,2,2,4,4]
FH= [3,2,5,1,3,1,7,4,4,10,7,2,5,4,10,2,2,8,4,7]
FMH= [45,42,34, 41,32,36,35,17,32,32,25,32,30,31,36,32,41,23,31,41]

```

## Sanity Test

```

In [6]: def interpret_ttest(a, b, alpha=0.05):
    t_stat, df, cv, p = independent_ttest(a, b, alpha)
    print('t=%.3f, df=%d, cv=%.5f, p=%.8f' % (t_stat, df, cv, p))
    # # interpret via critical value
    # if abs(t_stat) <= cv:
    #     print('Accept null hypothesis (Same distributions).')
    # else:

```

```
#         print('Reject the null hypothesis (Different distributions).')
# interpret via p-value
if p > alpha:
    print('Accept null hypothesis (Same distributins).')
else:
    print('Reject the null hypothesis (Different distributions).')
```

In [7]: `interpret_ttest(s,s2)`

```
t=48.100, df=9998, cv=1.64501, p=0.00000000
Reject the null hypothesis (Different distributions).
```

In [8]:

```
def interpret_kwtest(a, b, alpha = 0.05):
    stat, p = stats.kruskal(a, b)
    print('Statistics=%.3f, p=%.8f' % (stat, p))
    # interpret

    if p > alpha:
        print('Same distributions (fail to reject H0)')
    else:
        print('Different distributions (reject H0)')
```

In [9]: `interpret_kwtest(s,s2)`

```
Statistics=1889.883, p=0.00000000
Different distributions (reject H0)
```

In [10]:

```
# D'Agostino and Pearson's Test
from scipy.stats import normaltest
# normality test
# interpret results
def interpret_normaltest(data, alpha=0.05):
    stat, p = normaltest(data)
    print('Statistics=%.3f, p=%.3f' % (stat, p))

    if p > alpha:
        print('Sample looks Gaussian (fail to reject H0)')
    else:
        print('Sample does not look Gaussian (reject H0)')
```

In [11]: `interpret_normaltest(s)`

```
Statistics=0.871, p=0.647
Sample looks Gaussian (fail to reject H0)
```

In [12]: `interpret_normaltest(s2)`

```
Statistics=0.842, p=0.656
Sample looks Gaussian (fail to reject H0)
```

In [13]: `interpret_normaltest(zz)`

```
Statistics=466.626, p=0.000
Sample does not look Gaussian (reject H0)
```

## Normality Tests

In [14]: `distributions = {`

```
'F': F,
'FH': FH,
'FMH': FMH,
'FM': FM,
}
```

```
In [15]: for k, v in distributions.items():
          print(f'Distributions {k}')
          interpret_normaltest(v)
          print('\n\n')
```

```
Distributions F
Statistics=0.920, p=0.631
Sample looks Gaussian (fail to reject H0)
```

```
Distributions FH
Statistics=1.946, p=0.378
Sample looks Gaussian (fail to reject H0)
```

```
Distributions FMH
Statistics=1.675, p=0.433
Sample looks Gaussian (fail to reject H0)
```

```
Distributions FM
Statistics=7.643, p=0.022
Sample does not look Gaussian (reject H0)
```

## Comparison Scenarios

```
In [16]: scenarios_gauss = {
          'F-FH': (F, FH),
          'F-FMH': (F, FMH),
          'FH-FMH': (FH, FMH),
        }

          scenarios_non_parametric = {
          'F-FM': (F, FM),
          'FH-FM': (FH, FM),
          'FM-FMH': (FM, FMH),
        }
```

## Comparison for gaussian distributions - T-Test

```
In [17]: for k, v in scenarios_gauss.items():
          a, b = v
          print(f'Scenario {k}')
```

```
interpret_ttest(a,b)
print('\n\n')
```

Scenario F-FH  
t=-1.617, df=38, cv=1.68595, p=0.11420297  
Accept null hypothesis (Same distributins).

Scenario F-FMH  
t=-18.947, df=38, cv=1.68595, p=0.00000000  
Reject the null hypothesis (Different distributions).

Scenario FH-FMH  
t=-17.560, df=38, cv=1.68595, p=0.00000000  
Reject the null hypothesis (Different distributions).

## Non parametric test using KW-Test

```
In [18]: for k, v in scenarios_non_parametric.items():
          a, b = v
          print(f'Scenario {k}')
          interpret_kwtest(a,b)
          print('\n\n')
```

Scenario F-FM  
Statistics=0.401, p=0.52637675  
Same distributions (fail to reject H0)

Scenario FH-FM  
Statistics=3.266, p=0.07071398  
Same distributions (fail to reject H0)

Scenario FM-FMH  
Statistics=29.599, p=0.00000005  
Different distributions (reject H0)