

**School of Built Environment, Engineering and Computing**

****

**Week 2 Part Self Paced Tutorials Inferential Statistics 1**

**(T-Test and ANOVA)**

**By**

**Prof. Ah-Lian Kor**

**Inferential Statistics 1: T-Test**

T-test is for hypothesis testing.

In this tutorial, we shall conduct 3 types of t-tests:

1. An independent samples t-test which compares the means of 2 groups
2. A paired sample t-test
3. A one sample t-test which tests the mean of a single group against a known mean

There are three different tests:

1. Right-tail test
2. Left-tail test
3. Two-tail test

**Parametric Test**

Assumptions: normal distribution of values; equal (or non-equal) variances.

**Python:**

You can conduct your t-tests in pandas, scipy or numpy.

Reference: <https://scipy-lectures.org/packages/statistics/index.html#the-pandas-data-frame>

**Descriptive Statistics Tasks**

1. Download the brain\_size.csv file.
2. Examine the file: look at the dataset in brain\_size.csv. The delimiter or separator for the csv file is ; and missing values are replaced with .
3. History of the brain dataset and analyses conducted <https://www3.nd.edu/~busiforc/handouts/Data%20and%20Stories/correlation/Brain%20Size/brainsize.html>
4. Data Dictionary

|  |  |
| --- | --- |
| **Column Title** | **Description** |
| Gender | Male or Female |
| FSIQ | Full Scale IQ scores based on the four Wechsler (1981) subtests |
| VIQ | Verbal IQ scores based on the four Wechsler (1981) subtests |
| PIQ | Performance IQ scores based on the four Wechsler (1981) subtests |
| Weight | body weight in pounds |
| Height | height in inches |
| MRI\_Count | total pixel Count from the 18 MRI scans (index for brain size) |

1. Statistical analysis using Scipy <https://scipy-lectures.org/packages/statistics/index.html#the-pandas-data-frame> . The steps are as follows:
   1. Read in the dataset (csv file) and display the first five records. Types the following in Jupyter Notebook:

import pandas as pd

dataset = pd.read\_csv('brain\_size.csv', sep=';', na\_values=".")

dataset.head()

* 1. Descriptive Statistics for the dataset

import pandas as pd

dataset = pd.read\_csv('brain\_size.csv', sep=';', na\_values=".")

dataset.describe

Note: we do not want column 1 which does not make sense. Go to (c).

* 1. Choose descriptive statistics of specific columns. Type the following:

import pandas as pd

dataset = pd.read\_csv('brain\_size.csv', sep=';', na\_values=".")

dataset[["FSIQ", "VIQ", "Weight", "Height", "MRI\_Count"]].describe()

* 1. Choose descriptive statistics of specific columns for gender = female. Type the following:

import pandas as pd

dataset = pd.read\_csv('brain\_size.csv', sep=';', na\_values=".")

dataset[dataset['Gender'] == 'Female'][["FSIQ", "VIQ"]].describe()

* 1. Choose descriptive statistics for groups by male and female. Type the following:

import pandas as pd

dataset = pd.read\_csv('brain\_size.csv', sep=';', na\_values=".")

groupby\_gender = dataset.groupby('Gender')

groupby\_gender.describe()

Note: There are 56 horizontal columns and it is hard to capture as well as display on the screen. We shall convert the horizontal display to vertical display. Look at (f).

* 1. Change horizontal to vertical display. Types the following:

import pandas as pd

dataset = pd.read\_csv('brain\_size.csv', sep=';', na\_values=".")

groupby\_gender = dataset.groupby('Gender')

df1 = groupby\_gender.describe()

df\_transposed = df1.T

df\_transposed

* 1. If you want to run individual descriptive functions then type the following (including skewness and kurtosis)

import pandas as pd

dataset = pd.read\_csv('brain\_size.csv', sep=';', na\_values=".")

groupby\_gender = dataset.groupby('Gender')

print("Mean by Gender")

print(groupby\_gender.mean())

print("\n")

print("Median by Gender")

print(groupby\_gender.median())

print("\n")

print("Describe FSIQ and VIQ by Gender")

df1 = groupby\_gender[['FSIQ','VIQ']].describe()

df\_transposed = df1.T

print(df\_transposed)

print("\n")

print("Skewness by Gender")

df2 = groupby\_gender.skew()

df2\_transposed = df2.T

print(df2\_transposed)

print("\n")

print("Kurtosis by Gender")

df3 = pd.DataFrame(dataset)

print(df3.groupby('Gender').apply(pd.DataFrame.kurt))

print("\n")

* 1. Frequency count for male and female in the dataset for all the columns. Type the following:

import pandas as pd

dataset = pd.read\_csv('brain\_size.csv', sep=';', na\_values=".")

print('\n')

print('Number of Females in the Dataset')

print(dataset[dataset['Gender'] == 'Female'].count())

print('\n')

print('Number of Males in the Dataset')

print(dataset[dataset['Gender'] == 'Male'].count())

1. **Plotting Graphs**

**Look at Week 2 Tutorials**

Scatter Plot

Bar Charts

Line Graphs

Time Series Graphs

**Look at the following resource**:

<https://scipy-lectures.org/packages/statistics/index.html#the-pandas-data-frame>

1. **Inferential Statistics Tasks**

**Part 1: Parametric Tests – Assumption: Dataset has a normal distribution**

For parametric tests, we would assume that the dataset has a normal distribution.

**T-Test**

**Type 1: One-Sample T-test**

**Make sure you have brain\_size.csv file in your Jupyter Notebook DAV folder.**

**Example 1.1: You would like to compare the VIQ mean of a sample to a specific number or a population mean (the first example is 0).**

**Type the following:**

import pandas as pd

from scipy import stats

dataset = pd.read\_csv('brain\_size.csv', sep=';', na\_values=".")

stats.ttest\_1samp(dataset['VIQ'], 0)

**Displayed Output**

Ttest\_1sampResult(statistic=30.088099970849328, pvalue=1.3289196468728067e-28)

**Interpretation of the Output**

**Two values are found in the output:**

tvalue = 30.088 (3 decimal places)

pvalue = 1.3289196468728067e-28 (this means 1.3289196468728067 x 10 to the power of -28

**We can choose the following confidence level values:**

* + 1. Confidence Level (95%) or alpha value, α = 0.05
    2. Confidence Level (98%) or alpha value, α = 0.02
    3. Confidence Level (99%) or alpha value, α = 0.01

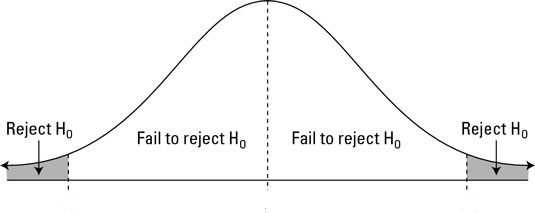
**We choose a two-tail t-test**

Null Hypothesis (H0): mean of VIQ = 0;

Alternative Hypothesis (Ha): mean of VIQ ≠ 0

**The distribution:**

1. **Choose:** Confidence Level (95%) or alpha value, α = 0.05
2. **It is a two-tailed test, therefore,** α/2 = 0.05/2 = 0.025
3. **The displayed p value is** 1.3289196468728067e-28 < α/2 = 0.025, therefore it falls in the Reject Ho region. Therefor the conclusion is: mean of VIQ is not equal to 0.



α/2 = 0.025

α/2 = 0.025

**Example 1.2 for 1-sample t-test**

The Wechsler **Adult** Intelligence Scale and the Stanford-Binet test, scores that fall between 90 and 109 are considered **average IQ** scores.

Choose a 2-tail t-test the following hypotheses:

Null Hypothesis (H0): mean of VIQ = 100;

Alternative Hypothesis (Ha): mean of VIQ ≠ 100

**Run the above hypothesis testing and provide an interpretation of the results for** α = 0.05; 0.02; 0.01.

**Resources**

<https://scipy-lectures.org/packages/statistics/index.html#the-pandas-data-frame>

**Type 2: Two-Sample T-Test**

**Make sure you have brain\_size.csv file in your Jupyter Notebook DAV folder.**

**Example 2.1: You would like to compare the means of Female FSIQ and Male FSIQ.**

**Type the following:**

import pandas as pd

from scipy import stats

dataset = pd.read\_csv('brain\_size.csv', sep=';', na\_values=".")

female\_fsiq = dataset[dataset['Gender'] == 'Female']['FSIQ']

male\_fsiq = dataset[dataset['Gender'] == 'Male']['FSIQ']

stats.ttest\_ind(female\_fsiq, male\_fsiq)

**Displayed Output**

Ttest\_indResult(statistic=-0.4026724743703011, pvalue=0.6894456253897778)

**Interpretation of the Output**

**Two values are found in the output:**

tvalue = -0.403 (3 decimal places)

pvalue = 0.689

**We can choose the following confidence level values:**

1. Confidence Level (95%) or alpha value, α = 0.05
2. Confidence Level (98%) or alpha value, α = 0.02
3. Confidence Level (99%) or alpha value, α = 0.01

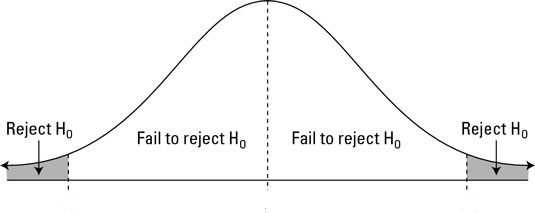
**We choose a two-tail t-test**

Null Hypothesis (H0): Female FSIQ mean = Male FSIQ mean;

Alternative Hypothesis (Ha): Female FSIQ mean ≠ Male FSIQ mean

**The distribution:**

1. **Choose:** Confidence Level (99%) or alpha value, α = 0.01
2. **It is a two-tailed test, therefore,** α/2 = 0.01/2 = 0.005
3. **The displayed p value is 0.689 >** α/2 = 0.005, therefore it falls in the Fail to Reject Ho ( or in other words, Accept Ho ) region. Therefor the conclusion is: there is no significant difference between the Female and Male FSIQ means.



α/2 = 0.005

α/2 = 0.005

**Example 2.2 for 2-sample t-test**

Choose a 2-tail t-test the following hypotheses:

Null Hypothesis (H0): Female PIQ mean = Male PIQ mean;

Alternative Hypothesis (Ha): Female PIQ mean Male ≠ PIQ mean;

**Run the above hypothesis testing and provide an interpretation of the results for** α = 0.05; 0.02; 0.01.

**Resources**

<https://scipy-lectures.org/packages/statistics/index.html#the-pandas-data-frame>

**Type 3: Matched Pair T-Test (or Repeated Measure of a Single Sample)**

**Make sure you have brain\_size.csv file in your Jupyter Notebook DAV folder.**

**Example 3.1: You would like to compare the means of VIQ and PIQ for the same individual.**

**Type the following:**

import pandas as pd

from scipy import stats

dataset = pd.read\_csv('brain\_size.csv', sep=';', na\_values=".")

stats.ttest\_rel(dataset['FSIQ'], dataset['PIQ'])

**Displayed Output**

Ttest\_relResult(statistic=1.7842019405859857, pvalue=0.08217263818364236)

**Interpretation of the Output**

**Two values are found in the output:**

tvalue = 1.784 (3 decimal places)

pvalue = 0.082

**We can choose the following confidence level values:**

1. Confidence Level (95%) or alpha value, α = 0.05
2. Confidence Level (98%) or alpha value, α = 0.02
3. Confidence Level (99%) or alpha value, α = 0.01

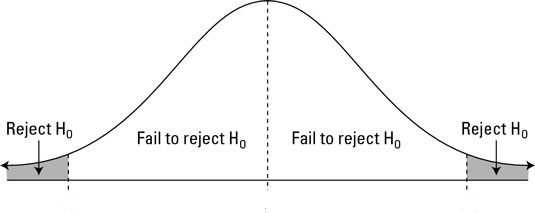
**We choose a two-tail t-test**

Null Hypothesis (H0): VIQ mean = PIQ mean (sometimes written as VIQ mean – PIQ mean = 0);

Alternative Hypothesis (Ha): VIQ mean ≠ PIQ mean (sometimes written as VIQ mean – PIQ mean ≠ 0)

**The distribution:**

1. **Choose:** Confidence Level (98%) or alpha value, α = 0.02
2. **It is a two-tailed test, therefore,** α/2 = 0.02/2 = 0.01
3. **The displayed p value is 0.082 >** α/2 = 0.01, therefore it falls in the Fail to Reject Ho ( or in other words, Accept Ho ) region. Therefor the conclusion is: there is no significant difference between the VIQ and PIQ means.



α/2 = 0.01

α/2 = 0.01

**Resources**

<https://scipy-lectures.org/packages/statistics/index.html#the-pandas-data-frame>

**Type 4: Two-Sample T-Test (Automatically general conclusion)**

**Make sure you have brain\_size.csv file in your Jupyter Notebook DAV folder.**

**Example 4.1: You would like to compare the means of Female FSIQ and Male FSIQ and automatically generate the conclusion.**

**Type the following:**

import pandas as pd

from scipy import stats

dataset = pd.read\_csv('brain\_size.csv', sep=';', na\_values=".")

female\_fsiq = dataset[dataset['Gender'] == 'Female']['FSIQ']

male\_fsiq = dataset[dataset['Gender'] == 'Male']['FSIQ']

t2, p = stats.ttest\_ind(female\_fsiq, male\_fsiq)

#two-tail 2-sample t-test

alpha\_half = 0.005 #alpha is 0.01 or level of confidence is 99%

print("p value = {:g}".format(p))

print("t value = {:g}". format(t2))

if p < alpha\_half: # null hypothesis: x comes from a normal distribution

print("The null hypothesis can be rejected")

else:

print("The null hypothesis is accepted")

**Displayed Output**

p value = 0.689446

t value = -0.402672

The null hypothesis is accepted

**Check the results with Example 2.1.**

1. **One-Way Analysis of Variance (ANOVA)**

**Assumptions for ANOVA Test (Parametric Test)** [**https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.f\_oneway.html#scipy.stats.f\_oneway**](https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.f_oneway.html#scipy.stats.f_oneway)

The ANOVA test has important assumptions that must be satisfied in order for the associated p-value to be valid.

1. The samples are independent.
2. Each sample is from a normally distributed population.
3. The population standard deviations of the groups are all equal. This property is known as homoscedasticity.

**Example 8.1 Extracted from** [**https://reneshbedre.github.io/blog/anova.html**](https://reneshbedre.github.io/blog/anova.html)

* 1. Download score.xlsx file and save it in Jupyter Notebook folder, DAV.
  2. Descriptive statistics and boxplot for the dataset. Type the following:

import pandas as pd

from matplotlib import pyplot as plt

dataset = pd.read\_excel("score.xlsx")

print(dataset.describe())

dataset.boxplot(column=['A', 'B', 'C', 'D'], grid=False)

plt.title("Subject Scores")

plt.xlabel("Subjects")

plt.ylabel("Score")

* 1. Conduct One-way ANOVA on score.xlsx. We would like to compare the means of Subjects A, B, C, and D. The two hypotheses are as follows:

**Note: for ANOVA, it is only one-tail test**

Null Hypothesis (H0): A mean = B mean = C mean = D mean (all the means of all the subjects are equal to each other);

Alternative Hypothesis (Ha): At least one of the means for A, B, C, and D is not equal to the other

**We can choose the following confidence level values:**

1. Confidence Level (95%) or alpha value, α = 0.05
2. Confidence Level (98%) or alpha value, α = 0.02
3. Confidence Level (99%) or alpha value, α = 0.01
   1. **Type the following codes for one way anova:**

import pandas as pd

import scipy.stats as stats

dataset = pd.read\_excel("score.xlsx")

fvalue, pvalue = stats.f\_oneway(dataset["A"], dataset["B"],dataset["C"], dataset["D"])

print("F Value = {:g} ".format(fvalue))

print("P Value = {:g} ".format(pvalue))

Note: output format details are found here: <https://www.python-course.eu/python3_formatted_output.php>

**The output is as follows:**

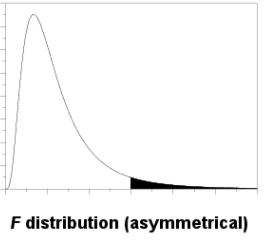
F Value = 17.4928

P Value = 2.63924e-05

**Interpretation:**

We choose α = 0.05.

The p value < α = 0.05 so we reject the null hypothesis. **The conclusion is that at least one mean for A, B, C, D is significantly different from the rest.**



α = 0.05

**Automate Conclusion**

**Type the following:**

import pandas as pd

import scipy.stats as stats

dataset = pd.read\_excel("score.xlsx")

fvalue, pvalue = stats.f\_oneway(dataset["A"], dataset["B"],dataset["C"], dataset["D"])

print("F Value = {:g} ".format(fvalue))

print("P Value = {:g} ".format(pvalue))

alpha = 0.05

if pvalue < alpha: # null hypothesis: x comes from a normal distribution

print("The null hypothesis can be rejected")

else:

print("The null hypothesis is accepted")

**Output**

F Value = 17.4928

P Value = 2.63924e-05

The null hypothesis can be rejected

**Compare the values and conclusion with the manual interpretation (shown above)**

**Example 8.2: One Way ANOVA**

Download the iris.csv file and save it in Jupyter Notebook folder, DAV.

1. Conduct descriptive statistical analysis on dataset in iris.csv. Type the following:

import pandas as pd

import scipy.stats as stats

dataset = pd.read\_csv("iris.csv")

groupby\_name = dataset.groupby('name')

print(groupby\_name.head())

print(groupby\_name['petal\_width'].describe())

print("\n")

print("\n")

df=groupby\_name.describe()

df\_transposed = df.T

print(df\_transposed)

1. You notice there are 3 types of irises by the name: setosa, versicolor, and vriginica.
2. We would like to test whether there is any significant difference in the mean of petal width for all the three types of irises. Here we shall use One-Way ANOVA.
   1. The two hypotheses are as follows:

**Note: for ANOVA, it is only one-tail test**

Null Hypothesis (H0): Setosa petal width mean = Versicolor petal width mean = Virginica petal width mean (all the petal width means for all the irises are equal to each other);

Alternative Hypothesis (Ha): At least one of the petal width means for the irises is not equal to the other

**We can choose the following confidence level values:**

1. Confidence Level (95%) or alpha value, α = 0.05
2. Confidence Level (98%) or alpha value, α = 0.02
3. Confidence Level (99%) or alpha value, α = 0.01
4. There are two sets of codes to compare the petal widths of the 3 irises. Type the following codes:

**Set 1**

import pandas as pd

import scipy.stats as stats

df = pd.read\_csv("iris.csv")

fvalue, pvalue = stats.f\_oneway(df['petal\_width'][df['name'] == 'setosa'],

df['petal\_width'][df['name'] == 'versicolor'],

df['petal\_width'][df['name'] == 'virginica'])

print("F Value = {:g} ".format(fvalue))

print("P Value = {:g} ".format(pvalue))

**Output**

F Value = 959.324

P Value = 4.37696e-85

**Interpretation**

Choose an alpha value and draw your conclusion. Just as in previous exercises.

**Set 2 – you store the data in arrays**

import pandas as pd

import scipy.stats as stats

df = pd.read\_csv("iris.csv")

df2 =(df['petal\_width'][df['name'] == 'setosa'],

df['petal\_width'][df['name'] == 'versicolor'],

df['petal\_width'][df['name'] == 'virginica'])

fvalue, pvalue = stats.f\_oneway(df2[0], df2[1], df2[2])

print("F Value = {:g} ".format(fvalue))

print("P Value = {:g} ".format(pvalue))

**Output**

F Value = 959.324

P Value = 4.37696e-85

**Interpretation**

Choose an alpha value and draw your conclusion. Just as in previous exercises.

**Example 8.3: One Way ANOVA Table**

Note, we shall use statsmodel and ANOVA linear model and OLS (Ordinary Least Square). We shall revisit these in detail when we conduct regression analysis. Download the rehab.csv file and save it in Jupyter Notebook DAV folder.

Type the following:

#https://www.statsmodels.org/stable/examples/notebooks/generated/interactions\_anova.html

import pandas as pd

import scipy.stats as stats

from matplotlib import pyplot as plt

import statsmodels.api as sm

from statsmodels.formula.api import ols

from statsmodels.stats.anova import anova\_lm

dataset = pd.read\_csv("rehab.csv")

#Plot box plots grouped by Fitness - Groups 1, 2, 3

fig, ax = plt.subplots(figsize=(8,6))

fig = dataset.boxplot('Time', 'Fitness', ax = ax, grid=False)

#Sum of Square

sum\_lm = ols('Time ~ Fitness', data=dataset).fit()

print(anova\_lm(sum\_lm))

print(anova\_lm(sum\_lm, typ=2))

print(anova\_lm(sum\_lm, typ=3))

Output

df sum\_sq mean\_sq F PR(>F)

Fitness 1.0 666.216867 666.216867 34.749543 0.000006

Residual 22.0 421.783133 19.171961 NaN NaN

sum\_sq df F PR(>F)

Fitness 666.216867 1.0 34.749543 0.000006

Residual 421.783133 22.0 NaN NaN

sum\_sq df F PR(>F)

Intercept 6679.707064 1.0 348.410223 5.611672e-15

Fitness 666.216867 1.0 34.749543 6.230589e-06

Residual 421.783133 22.0 NaN NaN

**Interpretation**

**Explanation:**

Focus on the F value and p value (this is for accepting or rejecting the null hypothesis – previously explained)

Explanation of ANOVA Table

<https://www.itl.nist.gov/div898/handbook/prc/section4/prc433.htm#:~:text=These%20mean%20squares%20are%20denoted,hypotheses%20about%20the%20population%20means>.

<https://www.itl.nist.gov/div898/handbook/ppc/section2/ppc231.htm>

**Explanation for Types 1, and 2 Errors**

1. Type 1 error is rejecting the null hypothesis when you shouldn’t have
2. Type 2 error is not rejecting null hypothesis when you should have

**Explanation for Types 1, 2, 3 ANOVA**

Very good explanation is given here <http://md.psych.bio.uni-goettingen.de/mv/unit/lm_cat/lm_cat_unbal_ss_explained.html>

**Example 8.4: Two-Way ANOVA**

**Download tooth\_growth.xlsx file and save to Jupyter Notebook DAV folder.**

**The dependent variable is the length of the tooth (LEN); 2 independent variables are the type of supplement(sup) and dosage (dose).**

#https://plotly.com/python/v3/anova/

#Two Way ANOVA

#len - length of tooth, supp - supplement, dose - dosage

import pandas as pd

import scipy.stats as stats

from matplotlib import pyplot as plt

import statsmodels.api as sm

from statsmodels.formula.api import ols

from statsmodels.stats.anova import anova\_lm

dataset = pd.read\_excel('tooth\_growth.xlsx')

print(dataset.head())

fig, ax = plt.subplots(figsize=(8,6))

fig = dataset.boxplot('len', 'supp', ax = ax, grid=False)

formula = 'len ~ supp + dose + supp \*dose'

model = ols(formula, dataset).fit()

anova\_table = sm.stats.anova\_lm(model, typ=2)

print("\n")

print(anova\_table)

**Output**

sum\_sq df F PR(>F)

supp 205.350000 1.0 12.317020 8.936452e-04

dose 2224.304298 1.0 133.415146 1.907985e-16

supp:dose 88.920107 1.0 5.333483 2.463136e-02

Residual 933.634929 56.0 NaN NaN

**Interpretation**

At a confidence level of 95% (or alpha = 0.05), the interaction between supplement and dose seems to have a significant effect on the length of the tooth (len) because the pvalue of 0.024 > alpha value of 0.05.

**Resources**

One-Way ANOVA <https://www.pythonfordatascience.org/anova-python/>

MANOVA <https://www.marsja.se/python-manova-made-easy-using-statsmodels/>

<https://reneshbedre.github.io/blog/anova.html>

<https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.f_oneway.html>

<http://scipy-lectures.org/packages/statistics/index.html>

<https://www.pythonfordatascience.org/anova-python/>

<http://scipy-lectures.org/packages/statistics/index.html>

<https://plotly.com/python/v3/anova/>

<https://stackoverflow.com/questions/25537399/anova-in-python-using-pandas-dataframe-with-statsmodels-or-scipy>

Manual <https://www.marsja.se/four-ways-to-conduct-one-way-anovas-using-python/>

1. **Non-Parametric Tests (Exercises will be given next week)**
   1. **Normality Tests** normally distributed (Shapiro-Wilks Test)
   2. **Levene Test Levene test** can be used to check the Homogeneity of variances when the data is not drawn from normal distribution.
   3. **Mann Whitney is the non parametric alternative to the two sample t test** [**https://scipy-lectures.org/packages/statistics/index.html#the-pandas-data-frame**](https://scipy-lectures.org/packages/statistics/index.html#the-pandas-data-frame)
   4. **Wilcoxon test is a nonparametric statistical test that compares two paired groups** [**https://scipy-lectures.org/packages/statistics/index.html#the-pandas-data-frame**](https://scipy-lectures.org/packages/statistics/index.html#the-pandas-data-frame)
   5. **Kruskal Willis test is the non parametric alternative to the One way ANOVA**
   6. **Friedman Repeated Tests** [**https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.kruskal.html#scipy.stats.kruskal**](https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.kruskal.html#scipy.stats.kruskal)

Meaning of na\_values

<https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.read_csv.html>

**Resources:**

<https://scipy-lectures.org/packages/statistics/index.html#the-pandas-data-frame>

<https://scipy-lectures.org/packages/statistics/index.html>

<https://pythonfordatascienceorg.wordpress.com/independent-t-test-python/>

<https://stackoverflow.com/questions/13404468/t-test-in-pandas>

<https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.ttest_ind.html>

<https://pythonfordatascienceorg.wordpress.com/paired-samples-t-test-python/>

<https://medium.com/swlh/hypothesis-testing-using-t-test-using-inferential-statistics-python-4dce44eb4146>

<https://www.pythonfordatascience.org/independent-samples-t-test-python/>

<https://towardsdatascience.com/inferential-statistics-series-t-test-using-numpy-2718f8f9bf2f>

<https://machinelearningmastery.com/how-to-code-the-students-t-test-from-scratch-in-python/>

<https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.ttest_ind.html>

<https://stackoverflow.com/questions/13404468/t-test-in-pandas>

**Seaborn – sttistical data visualisation**

<https://seaborn.pydata.org/>

User guide and tutorial <https://seaborn.pydata.org/tutorial.html>

<https://stackoverflow.com/questions/25537399/anova-in-python-using-pandas-dataframe-with-statsmodels-or-scipy>

<https://stackoverflow.com/questions/43149149/anova-in-python-using-scipy-without-manually-typing-in-every-group-name>

<https://www.youtube.com/watch?v=vvy_YRiEvpA>