Descripció: Aprèn a realitzar test d'hipòtesis amb Python.

Objectius:

Calcular el p-valor de diferents variable o conjunt de variables. Interpretar el p-valor i dir si rebutja la hipòtesi nul·la.

Durada: 3 dies

Lliurament: Enviar la URL a un repositori anomenat Hypothesis testing que contingui la solució. S'ha d'entregar cada Excercici en un mateix fitxer i en un repositori.

Nivell 1

- **Exercici 1:** Agafa un conjunt de dades de tema esportiu que t'agradi i selecciona un atribut del conjunt de dades. Calcula el p-valor i digues si rebutja la hipòtesi nul·la agafant un alfa de 5%.

In [1]:

```
import numpy as np
import pandas as pd
import scipy.stats as stats
from scipy.stats import ttest_ind
from scipy.stats import ttest_rel
```

https://dataverse.no/file.xhtml?persistentId=doi:10.18710/PJONBM/OEJGQM&version=1.0

Contenido de los datos

- Sport = which sport the athlete practices
- Sex = gender of the athlete
- Age = Age of the athlete
- Bodymass = weight in kg
- 10m_(s) = time mark at 10 meters
- 20m_(s) = time mark at 20 meters
- 30m_(s) = time mark at 30 meters
- 40m_(s) = time mark at 40 meters
- F0 = theoretical maximal horizontal force production as extrapolated from the linear sprint F-V relationship (Y-intercept of the linear F-V relationship). That is, maximal force output in the horizontal direction. Corresponds to the initial push of the athlete onto the ground during sprint acceleration. The higher the value, the higher the sprint-specific horizontal force production.
- V0 = theoretical maximal running velocity as extrapolated from the linear sprint F-V relationship (X-intercept of the linear F-V relationship). V0 is slightly higher than the actual maximal velocity. The theoretical maximal running velocity the athlete would be able to reach should mechanical resistances (i.e. internal and external) against movement be null. It also represents the capability to produce horizontal force at very high running velocities.

- Pmax = maximal mechanical power output in the horizontal direction, computed as Pmax = F0·V0/4, or as the apex of the P-V 2nd degree polynomial relationship. That is, the maximal power output capability of the athlete in the horizontal direction (per unit body mass) during sprint acceleration.

- FVSlope = Slope of the linear F-V relationship, computed as SFV = -F0/V0. Index of the athlete's individual balance between force and velocity capabilities. The steeper the slope, the more negative its value, the more "force-oriented" the F-V profile, and vice versa.

- RFmax = maximal ratio of force. Ratio of force is computed as the ratio of the step-averaged horizontal component of the ground reaction force to the corresponding resultant force. Maximal ratio of force is computed as maximal value of RF for sprint times > 0.3 s. This measure expresses the theoretically maximal effectiveness of force application. Direct measurement of the proportion of the total force proportion that is directed in the forward direction of motion at sprint start.

- DRF = rate of decrease in RF with increasing speed during sprint acceleration, computed as the slope of the linear RF-V relationship. Describes the athlete's capability to limit the inevitable decrease in mechanical effectiveness with increasing speed. In other words, it is an index of the ability to maintain a net horizontal force production despite increasing running velocity. The more negative the slope, the faster the loss of effectiveness of force application during acceleration, and vice versa.

In [2]:

```
data = pd.read_csv('Sprinttest_Olympiatoppen.txt', sep = '\t', index_col=0)
data
```

Out[2]:

	Sport	Sex	Age_(y)	Bodymass_(kg)	10m_(s)	20m_(s)	30m_(s)	40m_(s)	F0_(N/kg)	V
ID										
1	Alpine_skiing	M	21	77	2.02	3.27	4.46	5.64	8.69	
2	Alpine_skiing	M	23	81	2.04	3.29	4.44	5.65	8.08	
3	Alpine_skiing	M	23	81	2.10	3.37	4.60	5.79	8.02	
4	Alpine_skiing	M	27	82	2.05	3.32	4.49	5.67	8.13	
5	Alpine_skiing	M	20	83	2.11	3.42	4.63	5.83	7.80	
...	...	...	...	...	...	...	...	...	...	
662	Weight/powerlifting	M	19	89	2.02	3.31	4.53	5.73	8.97	
663	Weight/powerlifting	M	31	92	2.14	3.52	4.86	6.14	8.36	
664	Weight/powerlifting	M	17	103	2.23	3.61	4.95	6.25	7.31	
665	Weight/powerlifting	M	20	110	2.16	3.51	4.82	6.15	7.93	
666	Weight/powerlifting	M	18	146	2.17	3.55	4.86	6.09	7.78	

666 rows × 14 columns

In [3]:

```
data.describe()
```

Out[3]:

	Age_(y)	Bodymass_(kg)	10m_(s)	20m_(s)	30m_(s)	40m_(s)	F0_(N/kg)	V0_(m
count	666.000000	666.000000	666.000000	666.000000	666.000000	666.000000	666.000000	666.0000
mean	22.783784	74.262763	2.103664	3.425976	4.659685	5.879745	8.018559	8.535
std	4.266122	12.331085	0.104083	0.189667	0.283075	0.441450	0.653056	0.7224

	Age_(y)	Bodymass_(kg)	10m_(s)	20m_(s)	30m_(s)	40m_(s)	F0_(N/kg)	V0_(m
min	16.000000	50.000000	1.860000	2.960000	3.960000	0.500000	6.290000	6.660(
25%	19.000000	65.000000	2.030000	3.290000	4.450000	5.620000	7.540000	8.070(
50%	22.000000	72.000000	2.090000	3.400000	4.620000	5.825000	8.000000	8.570(
75%	26.000000	82.000000	2.170000	3.540000	4.840000	6.120000	8.457500	9.0175
max	39.000000	146.000000	2.470000	4.090000	5.640000	7.230000	10.010000	10.930(

In [64]:

```
# Con la descripción se observa un valor mínimo de 0.5 segundos en la marca de 40 metros.
# Se busca la fila que contiene dicha cifra y se puede determinar que ese dato es erroneo,
# ya que la marca de 40 metros no puede ser inferior a las anteriores, puesto que la medic
# es tiempo acumulado.
data.loc[data['40m_(s)'] == 0.50]
```

Out[64]:

	Sport	Sex	Age_(y)	Bodymass_(kg)	10m_(s)	20m_(s)	30m_(s)	40m_(s)	F0_(N/kg)	V0_(m/s)	PC
ID											
486	Soccer	W	26	57	2.11	3.48	4.74	0.5	8.11	8.17	

In [65]:

```
df = data.drop([486])
```

In [66]:

```
df.describe()
```

Out[66]:

	Age_(y)	Bodymass_(kg)	10m_(s)	20m_(s)	30m_(s)	40m_(s)	F0_(N/kg)	V0_(m
count	665.000000	665.000000	665.000000	665.000000	665.000000	665.000000	665.000000	665.000(
mean	22.778947	74.288722	2.103654	3.425895	4.659564	5.887835	8.018421	8.5365
std	4.267506	12.322142	0.104161	0.189798	0.283271	0.389255	0.653538	0.7228
min	16.000000	50.000000	1.860000	2.960000	3.960000	4.950000	6.290000	6.660(
25%	19.000000	65.000000	2.030000	3.290000	4.450000	5.620000	7.540000	8.070(
50%	22.000000	72.000000	2.090000	3.400000	4.620000	5.830000	8.000000	8.570(
75%	26.000000	82.000000	2.170000	3.540000	4.840000	6.120000	8.460000	9.020(
max	39.000000	146.000000	2.470000	4.090000	5.640000	7.230000	10.010000	10.930(

Hipotesis:

- Ho: Los depotistas de menor peso tienen la mayor velocidad de aceleración.
- H1: El peso no influye en la velocidad de aceleración.

In [67]:

```
stat, p = ttest_rel(df['Bodymass_(kg)'], df['V0_(m/s)'])
print('stat=%.3f, p=%.3f' % (stat, p))
if p > 0.05:
    print('No se puede rechazar la hipótesis')
else:
    print('Rechachada la hipòtesis')
```

```
stat=140.823, p=0.000
Rechachada la hipòtesis
```

La hipótesis original es rechazada, no hay relación entre el peso y la velocidad máxima.

Nivell 2

- **Exercici 2:** Continua amb el conjunt de dades de tema esportiu que t'agradi i selecciona dos altres atributs del conjunt de dades. Calcula els p-valors i digues si rebutgen la hipòtesi nul·la agafant un alfa de 5%.

- En este ejercicio queremos ver si los atletas masculinos que practican esqui alpino alcanzan una velocidad máxima mayor a las atletas femeninas

Ho: Los atletas masculinos de esqui aplino alcanzan mayor velocidad máxima.

H1: el género no influye en la velocidad máxima a alcanzar.

In [68]:
```
data
```

Out[68]:

	Sport	Sex	Age_(y)	Bodymass_(kg)	10m_(s)	20m_(s)	30m_(s)	40m_(s)	F0_(N/kg)	V(
ID										
1	Alpine_skiing	M	21	77	2.02	3.27	4.46	5.64	8.69	
2	Alpine_skiing	M	23	81	2.04	3.29	4.44	5.65	8.08	
3	Alpine_skiing	M	23	81	2.10	3.37	4.60	5.79	8.02	
4	Alpine_skiing	M	27	82	2.05	3.32	4.49	5.67	8.13	
5	Alpine_skiing	M	20	83	2.11	3.42	4.63	5.83	7.80	
...	...	...	...	...	...	...	...	...	...	
662	Weight/powerlifting	M	19	89	2.02	3.31	4.53	5.73	8.97	
663	Weight/powerlifting	M	31	92	2.14	3.52	4.86	6.14	8.36	
664	Weight/powerlifting	M	17	103	2.23	3.61	4.95	6.25	7.31	
665	Weight/powerlifting	M	20	110	2.16	3.51	4.82	6.15	7.93	
666	Weight/powerlifting	M	18	146	2.17	3.55	4.86	6.09	7.78	

666 rows × 14 columns

In [69]:
```
# Se crea un nuevo set que incluya el sky alpino con los deportistas masculinos.
# Eliminamos las 3 primeras muestrar para igual muestra
male_alpine_skiing = data[ (data['Sport'] == 'Alpine_skiing') & (data['Sex'] == "M")]
male_alpine_skiing = male_alpine_skiing.iloc[3:]
male_alpine_skiing
```

Out[69]:

	Sport	Sex	Age_(y)	Bodymass_(kg)	10m_(s)	20m_(s)	30m_(s)	40m_(s)	F0_(N/kg)	V0_(m/s)
ID										
4	Alpine_skiing	M	27	82	2.05	3.32	4.49	5.67	8.13	8.97
5	Alpine_skiing	M	20	83	2.11	3.42	4.63	5.83	7.80	8.63
6	Alpine_skiing	M	21	85	1.98	3.17	4.34	5.45	8.96	9.22
7	Alpine_skiing	M	21	85	2.04	3.30	4.43	5.59	7.97	9.27

	Sport	Sex	Age_(y)	Bodymass_(kg)	10m_(s)	20m_(s)	30m_(s)	40m_(s)	F0_(N/kg)	V0_(m/s)
ID										
8	Alpine_skiing	M	24	85	1.99	3.24	4.39	5.51	8.72	9.09
9	Alpine_skiing	M	30	86	2.14	3.45	4.65	5.86	7.31	8.82
10	Alpine_skiing	M	30	86	2.03	3.23	4.33	5.43	7.93	9.77
11	Alpine_skiing	M	26	87	2.01	3.25	4.41	5.55	8.53	9.11
12	Alpine_skiing	M	31	87	2.02	3.31	4.53	5.72	8.98	8.49
13	Alpine_skiing	M	26	92	2.04	3.34	4.53	5.68	8.49	8.65

In [70]:

```
female_alpine_skiing = data[ (data['Sport'] == 'Alpine_skiing') & (data['Sex'] == "W")]
female_alpine_skiing
```

Out[70]:

	Sport	Sex	Age_(y)	Bodymass_(kg)	10m_(s)	20m_(s)	30m_(s)	40m_(s)	F0_(N/kg)	V0_(m/s)
ID										
14	Alpine_skiing	W	19	56	2.19	3.66	5.04	6.47	8.16	7.36
15	Alpine_skiing	W	23	64	2.19	3.58	4.88	6.20	7.42	8.02
16	Alpine_skiing	W	23	65	2.22	3.69	5.03	6.40	7.47	7.58
17	Alpine_skiing	W	19	65	2.11	3.44	4.67	5.95	7.87	8.48
18	Alpine_skiing	W	19	67	2.20	3.63	5.02	6.39	8.00	7.46
19	Alpine_skiing	W	29	67	2.19	3.64	4.95	6.32	7.59	7.74
20	Alpine_skiing	W	21	68	2.13	3.47	4.72	5.97	7.83	8.32
21	Alpine_skiing	W	26	68	2.29	3.75	5.14	6.51	7.00	7.50
22	Alpine_skiing	W	24	70	2.20	3.64	5.00	6.40	7.75	7.56
23	Alpine_skiing	W	23	71	2.23	3.65	5.00	6.34	7.39	7.70

In [71]:

```
stat, p = ttest_ind(male_alpine_skiing['V0_(m/s)'], female_alpine_skiing['V0_(m/s)'])
print('stat=%.3f, p=%.3f' % (stat, p))
if p > 0.05:
    print('No se puede rechazar la hipótesis')
else:
    print('Rechachada la hipòtesis')
```

```
stat=7.266, p=0.000
Rechachada la hipòtesis
```

Nivell 3

- **Exercici 3:** Continua amb el conjunt de dades de tema esportiu que t'agradi i selecciona tres atributs del conjunt de dades. Calcula el p-valor i digues si rebutja la hipòtesi nul·la agafant un alfa de 5%.

In [72]:

```
data['Sport'].unique()
```

Out[72]:

```
array(['Alpine_skiing', 'Athletics_jumping', 'Athletics_sprinting',
       'Athletics_throwing', 'Bandy', 'Basket', 'Beach/volleyball',
       'Bobsleigh', 'Combat_sports', 'Cross_country_skiing', 'Fencing',
       'Handball', 'Ice_hockey', 'Mogul_skiing', 'Nordic_combined',
       'Ski_jumping', 'Snowboard', 'Soccer', 'Speed_skating',
       'Table_tennis', 'Telemark_skiing', 'Tennis', 'Weight/powerlifting'],
      dtype=object)
```

In [73]:

```
data['V0_(m/s)'].mean()
```

Out[73]:

```
8.535975975975967
```

In [74]:

```
data['Bodymass_(kg)'].mean()
```

Out[74]:

```
74.26276276276276
```