

## ▼ Actividad: Diferencia entre medias poblacionales

Joel Isaías Solano Ocampo | A01639289

---

### ▼ Problema 1:

Un científico de datos está analizando los niveles de sodio en dos lotes diferentes de un mismo producto. El científico quiere determinar si los niveles de sodio son iguales para ambos lotes, por lo tanto, recabó las siguientes dos muestras de datos

- Lote A - (número de muestras 15): Nivel de Sodio (mg) -> 180, 160, 170, 190, 200, 175, 185, 195, 180, 170, 190, 185, 200, 175, 165.
- Lote B - (número de muestras 20): Nivel de Sodio (mg) -> 210, 215, 220, 225, 230, 215, 220, 225, 230, 235, 210, 215, 220, 225, 230, 215, 220, 225, 230, 23.

La varianza poblacional para el lote A es 57.05 y para el lote B 34.63. Con un nivel de confianza del 85% determina si ambos lotes tienen en promedio el mismo nivel de sodio.

1. Importamos librerías:

```
import pandas as pd
from scipy import stats
```

2. Agrupamos los datos de los lotes A y B en su respectiva lista de datos:

```
lote_a = [180, 160, 170, 190, 200, 175, 185, 195, 180, 170, 190, 185, 200, 175, 165]
lote_b = [210, 215, 220, 225, 230, 215, 220, 225, 230, 235, 210, 215, 220, 225, 230, 215, 220, 225, 230, 23]
```

3. Obtenemos el tamaño de los lotes A y B correspondientes:

```
n_a = len(lote_a)
n_b = len(lote_b)
```

4. Asignamos la varianza poblacional de cada lote proporcionado por el problema:

```
var_pob_a = 57.05
var_pob_b = 34.63
```

5. Obtenemos el P-valor con t-student para saber si rechazamos o no la hipótesis nula:

```
t_statistic, p_value = stats.ttest_ind_from_stats(mean1=sum(lote_a)/n_a, std1=(var_pob_a**0.5), nobs1=n_a, mean2=sum(lote_b)/n_b, std2=(var_pob_b**0.5), nobs2=n_b)
```

6. Llamamos a la función `hypotesis_rejected()` y le mandamos como parámetro el P-valor:

```
def hypotesis_rejected(p_v):
    if p_v < (1 - 0.85):
        print("Nivel de confianza: %d%% | Se puede rechazar la hipótesis nula." % (85))
    else:
        print("Nivel de confianza: %d%% | No se puede rechazar la hipótesis nula." % (85))

hypotesis_rejected(p_value)

Nivel de confianza: 85% | Se puede rechazar la hipótesis nula.
```

### ▼ Problema 4:

Se han tomado dos muestras del número de días que tardan los egresados de las universidades A y B en encontrar trabajo

- Universidad A: -> 180. 200. 190. 210. 175. 185. 195. 180. 205. 190. 200. 185. 210. 175. 195

- Universidad B: -> 210, 215, 220, 225, 230, 215, 220, 225, 230, 235, 210, 215, 220, 225, 230, 235

Utilizando un nivel de confianza del 95%, determina si hay evidencia estadística suficiente para concluir que existe una diferencia significativa en el tiempo promedio de búsqueda de empleo entre ambas universidades.

### 1. Importamos librerías:

```
import pandas as pd
from scipy import stats
```

### 2. Agrupamos los datos de las universidades A y B en su respectiva lista de datos:

**Nota:** Por alguna razón las variables no se pueden nombrar como uni\_a y uni\_b respectivamente ya que al aplicar el t-student no se reconoce a la variable por alguna razón, quizá por nomenclatura, por lo que ambos conjuntos de datos solo se nombraron como a y b.

```
a = [180, 200, 190, 210, 175, 185, 195, 180, 205, 190, 200, 185, 210, 175, 195]
b = [210, 215, 220, 225, 230, 215, 220, 225, 230, 235, 210, 215, 220, 225, 230, 235]
```

### 3. Obtenemos el tamaño de los lotes A y B correspondientes:

```
n_a = len(a)
n_b = len(b)
```

### 5. Obtenemos el P-valor con t-student para saber si rechazamos o no la hipótesis nula:

```
t_statistic, p_value = stats.ttest_ind(a=a, b=b, equal_var=False)
```

### 6. Llamamos a la función `hypotesis_rejected()` y le mandamos como parametro el P-valor:

```
def hypotesis_rejected(p_v):
    if p_v < (1 - 0.95):
        print("Nivel de confianza: %d%% | Se puede rechazar la hipótesis nula." % (95))
    else:
        print("Nivel de confianza: %d%% | No se puede rechazar la hipótesis nula." % (95))

hypotesis_rejected(p_value)

Nivel de confianza: 95% | Se puede rechazar la hipótesis nula.
```