Instituto Tecnológico y de Estudios Superiores de Monterrey

Intervalos de confianza

TC3004B.104 Inteligencia Artificial Avanzada para la Ciencia de Datos I

Profesores:

Ivan Mauricio Amaya Contreras

Blanca Rosa Ruiz Hernandez

Antonio Carlos Bento

Frumencio Olivas Alvarez

Hugo Terashima Marín

Julian Lawrence Gil Soares - A00832272

23 de Agosto de 2023

De la intensidad podemos inferir que las medias de ambos grupos están en un rango similar así que tener el marcapasos o no tener el marcapasos causa muy poca variación en la intensidad del pulso.

In [ ]:

```
from google.colab import drive
drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

In [ ]:

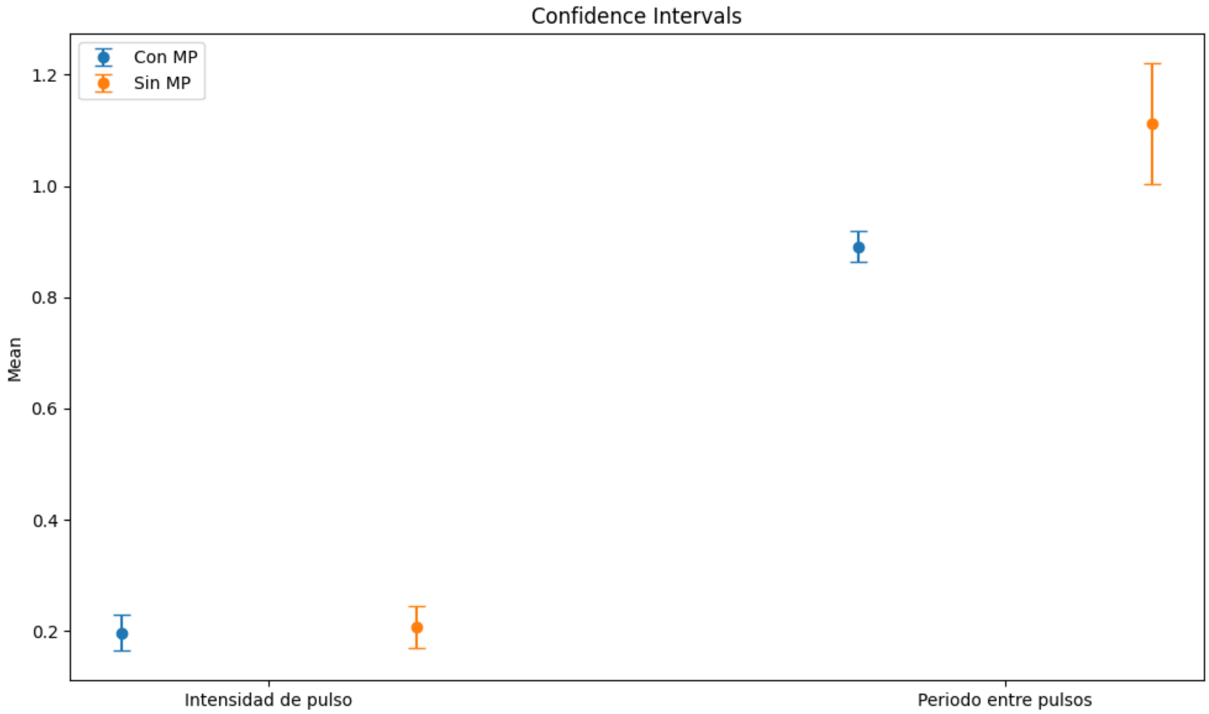
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from scipy import stats

# Load data
data = pd.read_csv("/content/drive/MyDrive/Stats/El marcapasos.csv")
```

In [ ]:

```
# Filter data
data_con_mp = data[data['Marcapasos'] == 'Con MP']
data_sin_mp = data[data['Marcapasos'] == 'Sin MP']

# Calculate means and standard errors
variables = ['Intensidad de pulso', 'Periodo entre pulsos']

means_con_mp = data_con_mp[variables].mean()
std_errors_con_mp = data_con_mp[variables].std() / np.sqrt(len(data_con_mp))

means_sin_mp = data_sin_mp[variables].mean()
std_errors_sin_mp = data_sin_mp[variables].std() / np.sqrt(len(data_sin_mp))

# Calculate t-distribution critical value
conf_level = 0.95
t_critical = stats.t.ppf((1 + conf_level) / 2, df=len(data_con_mp) - 1)

conf_intervals_con_mp = [(mean - t_critical * std_err, mean + t_critical * std_err)
                         for mean, std_err in zip(means_con_mp, std_errors_con_mp)]

conf_intervals_sin_mp = [(mean - t_critical * std_err, mean + t_critical * std_err)
                         for mean, std_err in zip(means_sin_mp, std_errors_sin_mp)]

# Display confidence intervals numerically

for variable, interval in zip(variables, conf_intervals_con_mp):
    print(f"{variable}: {interval}")

for variable, interval in zip(variables, conf_intervals_sin_mp):
    print(f"{variable}: {interval}")

# Graphing
plt.figure(figsize=(10, 6))

x_values = np.arange(len(variables))

plt.errorbar(x=x_values - 0.2, y=means_con_mp, yerr=std_errors_con_mp * t_critical,
             fmt='o', label='Con MP', capsize=5)

plt.errorbar(x=x_values + 0.2, y=means_sin_mp, yerr=std_errors_sin_mp * t_critical,
             fmt='o', label='Sin MP', capsize=5)
```

```
plt.xticks(x_values, variables)
plt.ylabel('Mean')
plt.title('Confidence Intervals')
plt.legend()
plt.tight_layout()
plt.show()
```

```
Intensidad de pulso: (0.16380352124059191, 0.22807883170058466)
Periodo entre pulsos: (0.8637940802137694, 0.9185588609627013)
Intensidad de pulso: (0.1699299679586898, 0.2442661104726827)
Periodo entre pulsos: (1.0028865670425062, 1.2206428447221997)
```



In [ ]: