

## ▼ Actividad 5 - Pruebas de Hipótesis

Frida Cano Falcón - A01752953

```
1 from google.colab import drive
2 drive.mount('/content/drive')
3 %cd "/content/drive/MyDrive/7mo Semestre/Estadistica"
4 !ls
```

```
Mounted at /content/drive
/content/drive/MyDrive/Semestres/7mo Semestre/Estadistica
Act1_Distribuciones_FridaCano_A01752953.ipynb
Act2_ExplorandoBases_FridaCano_A01752953.ipynb
Act3_Transformaciones_FridaCanoFalcon_A01752953.ipynb
Act4_IntervalosConfianza_FridaCanoFalcon_A01752953.ipynb
Act5_PruebasHipotesis_FridaCanoFalcon_A01752953.ipynb
'El marcapasos.csv'
mc-donalds-menu-1.csv
precios_autos.csv
```

```
1 import pandas as pd
2 import matplotlib.pyplot as plt
3 import numpy as np
4 from scipy import stats
5 from scipy.stats import t
6 # Importing library
7 from scipy.stats import skew, kurtosis
```

## ▼ Problema Enlatados

Prueba 1 - Hipótesis

$$H_0 : \mu = 11.7$$

$$H_1 : \mu \neq 11.7$$

## ▼ Prueba 2 - Regla de decisión

En el problema nos indican el nivel de confianza  $1 - \alpha = 0.98$ , entonces  $\alpha = 0.02$ .

Utilizaremos una *tdeStudent* porque no conocemos el valor de  $\sigma$  (desviación estándar poblacional) y el tamaño de muestra es pequeño ( $n < 30$ ).

Para calcular el valor de  $t_0$  de la region de rechazo:

```
1 alfa = 0.02
2 n = 21 # Tamaño de la muestra
3 t0 = t.ppf(alfa / 2, n - 1) # alfa se divide entre 2 porque es una prueba de dos colas
4 print("t0 =", t0)
```

```
t0 = -2.5279770027405464
```

Enotnces se rechaza  $H_0$  si:

- El valor  $p < \alpha \leq 0.02$
- Si  $|t^*| > |t_0|$ , *esdecir*,  $|t^*| > 2.528$

## ▼ Paso 3 - Análisis muestra

Para analizar la muestra, necesitamos calcular  $t^*$  y el valor de  $p$ .

```
1 from scipy import stats
2
3 X = np.array([11, 11.6, 11.6, 11.7, 10.9, 11.6, 12, 11.2, 11.5, 12, 12, 11.4, 11.2, 10.8, 10.5, 11.8, 12.2, 10.9, 11.8, 11.4, 12.1])
4 m = np.mean(X)
5 s = np.std(X, ddof=1) # Usamos ddof=1 para calcular la desviación estándar muestral
6 n = len(X)
7
8 print("La media de la muestra es", m)
9 print("La desviación de la muestra es", s)
10
11 sm = s / np.sqrt(n) # Desviación estándar de la media de la muestra (error estándar)
```

```

12 print("La desviación estándar de la media de la muestra (error estándar) es", sm)
13
14 te = (m - 11.7) / sm
15 print("\nt* =", te)
16
17 valor_p = 2 * stats.t.cdf(te, df=n-1)
18 print("Valor p =", valor_p)

```

La media de la muestra es 11.485714285714286  
 La desviación de la muestra es 0.4746427227775795  
 La desviación estándar de la media de la muestra (error estándar) es 0.10357553358039041

t\* = -2.068883517934234  
 Valor p = 0.051729895013436986

```

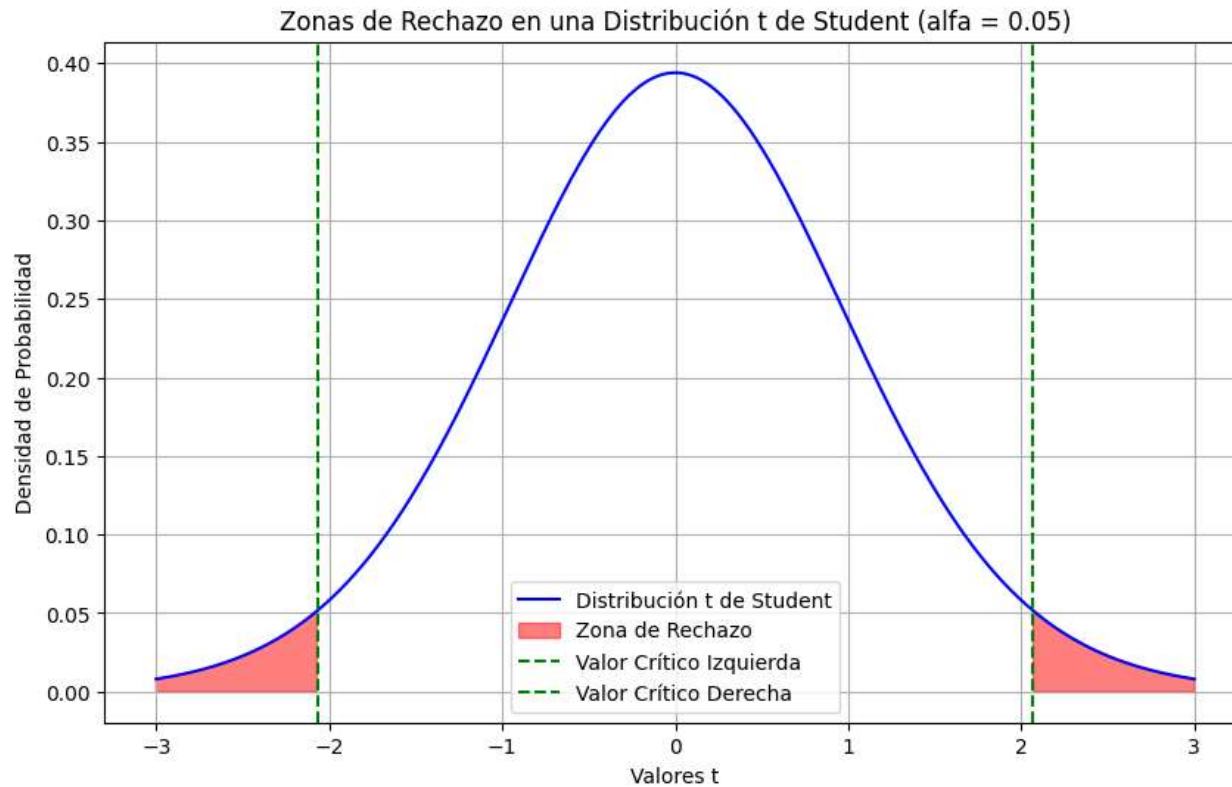
1 import numpy as np
2 import matplotlib.pyplot as plt
3 from scipy.stats import t
4
5 # Nivel de significancia (alfa)
6 alfa = valor_p
7
8 # Grados de libertad (puedes ajustar esto según tu prueba)
9 grados_libertad = 20
10
11 # Crear un rango de valores para la distribución t de Student
12 x = np.linspace(-3, 3, 1000)
13
14 # Calcular el valor crítico de la izquierda y derecha
15 valor_critico_izquierda = t.ppf(alfa / 2, df=grados_libertad)
16 valor_critico_derecha = t.ppf(1 - alfa / 2, df=grados_libertad)
17
18 # Crear el gráfico de la distribución t de Student
19 plt.figure(figsize=(10, 6))
20 plt.plot(x, t.pdf(x, df=grados_libertad), 'b-', label='Distribución t de Student')
21
22 # Colorear las zonas de rechazo
23 plt.fill_between(x, 0, t.pdf(x, df=grados_libertad), where=(x < valor_critico_izquierda) | (x > valor_critico_derecha), color='red', alpha=0.5, label='Zonas de Rechazo')
24
25 # Etiquetas y título
26 plt.xlabel('Valores t')
27 plt.ylabel('Densidad de Probabilidad')
28 plt.title('Zonas de Rechazo en una Distribución t de Student (alfa = 0.05)')
29
30 # Líneas verticales para los valores críticos

```

```

31 plt.axvline(valor_critico_izquierda, color='green', linestyle='--', label='Valor Crítico Izquierda')
32 plt.axvline(valor_critico_derecha, color='green', linestyle='--', label='Valor Crítico Derecha')
33
34 # Leyenda
35 plt.legend()
36
37 # Mostrar el gráfico
38 plt.grid(True)
39 plt.show()
40

```



## Conclusión

No se rechaza  $H_0$  porque:

- El valor  $\alpha < p \leq 0.0517$
- Si  $|t^*| < |t_0| = 2.528$

## ▼ Problema 2

Fowle Marketing Research, Inc., basa los cargos a un cliente bajo el supuesto de que las encuestas telefónicas (para recopilación de datos) pueden completarse en un tiempo medio de 15 minutos o menos. Si el tiempo es mayor a 15 minutos entonces se cobra una tarifa adicional. Compañías que contratan estos servicios piensan que el tiempo promedio es mayor a lo que especifica Fowle Marketing Research Inc. así que realizan su propio estudio en una muestra aleatoria de llamadas telefónicas y encuentran los siguientes datos:

Tiempo: 17, 11, 12, 23, 20, 23, 15, 16, 23, 22, 18, 23, 25, 14, 12, 12, 20, 18, 12, 19, 11, 11, 20, 21, 11, 18, 14, 13, 13, 19, 16, 10, 22, 18, 23

Por experiencias anteriores, se sabe que  $\sigma=4$  minutos. Usando un nivel de significación de 0.07, ¿está justificada la tarifa adicional?

Muestra tu procedimiento siguiendo los 4 pasos de solución Grafica la regla de decisión y el valor del estadístico de prueba. Concluye en el contexto del problema

## ▼ Prueba 1 - Hipótesis

$$H_0 : \mu = 15$$

$$H_1 : \mu \neq 15$$

```
1 X = np.array([17, 11, 12, 23, 20, 23, 15, 16, 23, 22, 18, 23, 25, 14, 12, 12, 20, 18, 12, 19, 11, 11, 20, 21, 11, 18, 14, 13, 13, 19, 16, 10, 22, 18, 23])
2 m = np.mean(X)
3 s = np.std(X, ddof=1) # Usamos ddof=1 para calcular la desviación estándar muestral
4 alfa = 0.07
5 n = len(X)
6 t0 = t.ppf(alfa / 2, n - 1) # alfa se divide entre 2 porque es una prueba de dos colas
```

```

7 t0 = abs(t0)
8 print("t0 =", t0)
9
10 print("La media de la muestra es", m)
11 print("La desviación de la muestra es", s)
12
13 sm = s / np.sqrt(n) # Desviación estándar de la media de la muestra (error estándar)
14 print("La desviación estándar de la media de la muestra (error estándar) es", sm)
15
16 te = (m - 11.7) / sm
17 print("\nt* =", te)
18
19 valor_p = 2 * stats.t.cdf(te, df=n-1)
20 print("Valor p =", valor_p)

t0 = 1.8708022084290248
La media de la muestra es 17.0
La desviación de la muestra es 4.530939390976874
La desviación estándar de la media de la muestra (error estándar) es 0.7658685408402277

t* = 6.9202476892253815
Valor p = 1.9999999435669558

```

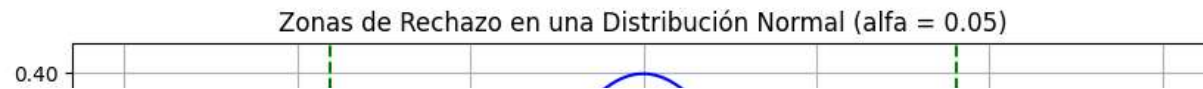
```

1 from scipy.stats import norm
2
3 # Nivel de significancia (alfa)
4 alfa = 0.07
5
6 # Crear un rango de valores para la distribución normal estándar
7 x = np.linspace(-3, 3, 1000)
8
9 # Calcular el valor crítico de la izquierda y derecha
10 valor_critico_izquierda = norm.ppf(alfa / 2)
11 valor_critico_derecha = norm.ppf(1 - alfa / 2)
12
13 # Crear el gráfico de la distribución normal estándar
14 plt.figure(figsize=(10, 6))
15 plt.plot(x, norm.pdf(x), 'b-', label='Distribución Normal Estándar')
16
17 # Colorear las zonas de rechazo
18 plt.fill_between(x, 0, norm.pdf(x), where=(x < valor_critico_izquierda) | (x > valor_critico_derecha), color='red', alpha=0.5, label='Zona de Rechazo')
19
20 # Etiquetas y título
21 plt.xlabel('Valores Z')
22 plt.ylabel('Densidad de Probabilidad')

```

```
23 plt.title('Zonas de Rechazo en una Distribución Normal (alfa = 0.05)')
24
25 # Líneas verticales para los valores críticos
26 plt.axvline(valor_critico_izquierda, color='green', linestyle='--', label='Valor Crítico Izquierda')
27 plt.axvline(valor_critico_derecha, color='green', linestyle='--', label='Valor Crítico Derecha')
28
29 # Leyenda
30 plt.legend()
31
32 # Mostrar el gráfico
33 plt.grid(True)
34 plt.show()
35
```





## Conclusión

Se rechaza  $H_0$  porque:

- El valor  $\alpha > p = 1.99$
- $|t_0| < |t^*| = 6.9$

