

```
In [1]: import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
import csv
```

```
In [2]: sujeto,ojo = [1,2,3,4,5,6,7,8,9,10],['D','I']
```

```
In [3]: Error_x,Error_y,Error_fijacion_x,Error_fijacion_y = [],[],[],[]
for i in range(9):
    for j in range(2):
        prueba = pd.read_csv(f'Prueba_error_{sujeto[i]}{ojo[j]}.csv',delimiter=';')
        for k in range(40):
            Error_x.append(prueba['Error X'][k])
            Error_y.append(prueba['Error Y'][k])
            Error_fijacion_x.append(prueba['Error Fijacion X'][k])
            Error_fijacion_y.append(prueba['Error Fijacion Y'][k])
```

```
In [4]: column_names = ['Error estimulo','Error Fijacion']
with open('Análisis de error.csv', mode='w', newline='') as file:
    writer = csv.writer(file, delimiter=';')
    writer.writerow(column_names)
    for row in zip(Error_x,Error_y,Error_fijacion_x,Error_fijacion_y):
        # Formatear cada número con dos decimales y usar el punto como separador
        formatted_row = [f'{value:.2f}' for value in row]
        writer.writerow(formatted_row)
```

Histogramas

```
In [9]: Columnas = [Error_x,Error_y,Error_fijacion_x,Error_fijacion_y]
nombre_columna = ['Error_x','Error_y','Error_fijacion_x','Error_fijacion_y']
for i in range(4):
    median_value,std_dev = np.median(np.sort(Columnas[i])),np.std(Columnas[i])
    print(f'Mediana en {nombre_columna[i]}:', median_value)
    print(f'Desviación Estandar en {nombre_columna[i]}:', np.std(Columnas[i]))
    # Crear el histograma
    plt.figure(figsize=(10,5))
    # Definir el ancho de los bins
    bin_width = 5
    # Calcular los bordes de los bins
    min_val = np.min(Columnas[i])
    max_val = np.max(Columnas[i])
    bin_edges = np.arange(min_val, max_val + bin_width, bin_width)
    plt.hist(Columnas[i], bins=bin_edges, edgecolor='black', alpha=0.7, color='blue')

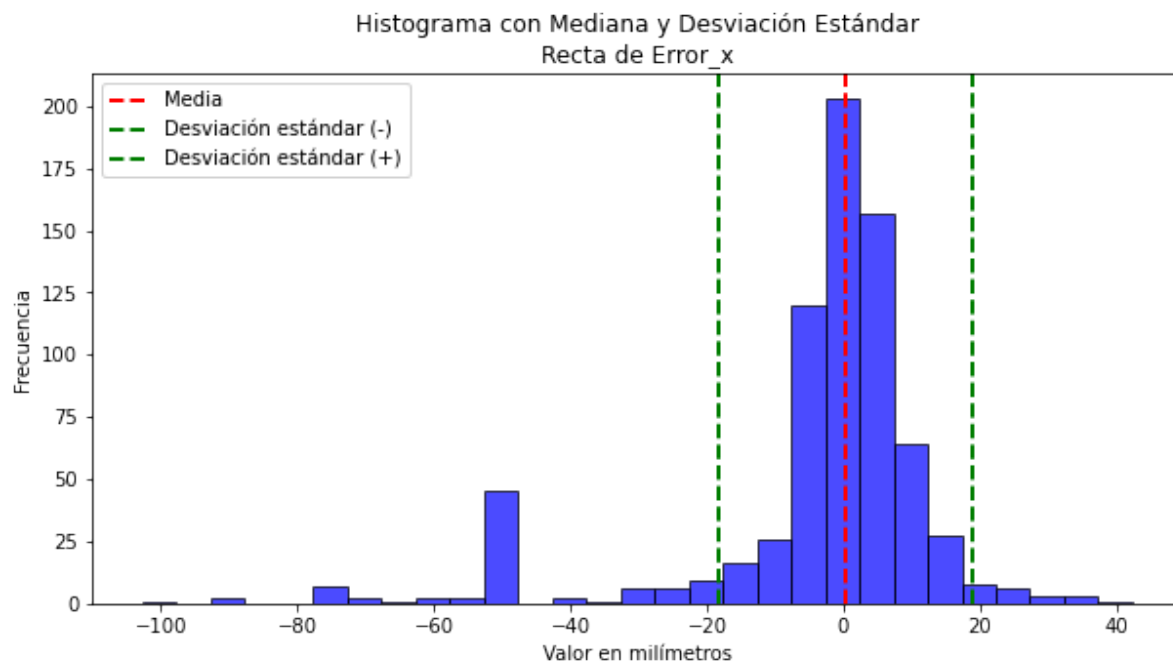
    # Añadir líneas verticales para el valor medio y la desviación estándar
    plt.axvline(median_value, color='red', linestyle='dashed', linewidth=2, label='Mediana')
    plt.axvline(median_value - std_dev, color='green', linestyle='dashed', linewidth=2, label='Desviación Estándar -')
    plt.axvline(median_value + std_dev, color='green', linestyle='dashed', linewidth=2, label='Desviación Estándar +')

    # Añadir títulos y etiquetas
    plt.title(f'Histograma con Mediana y Desviación Estándar\nRecta de {nombre_columna[i]}')
    plt.xlabel('Valor en milímetros')
    plt.ylabel('Frecuencia')
    plt.legend()

    # Mostrar el histograma
    plt.show()
```

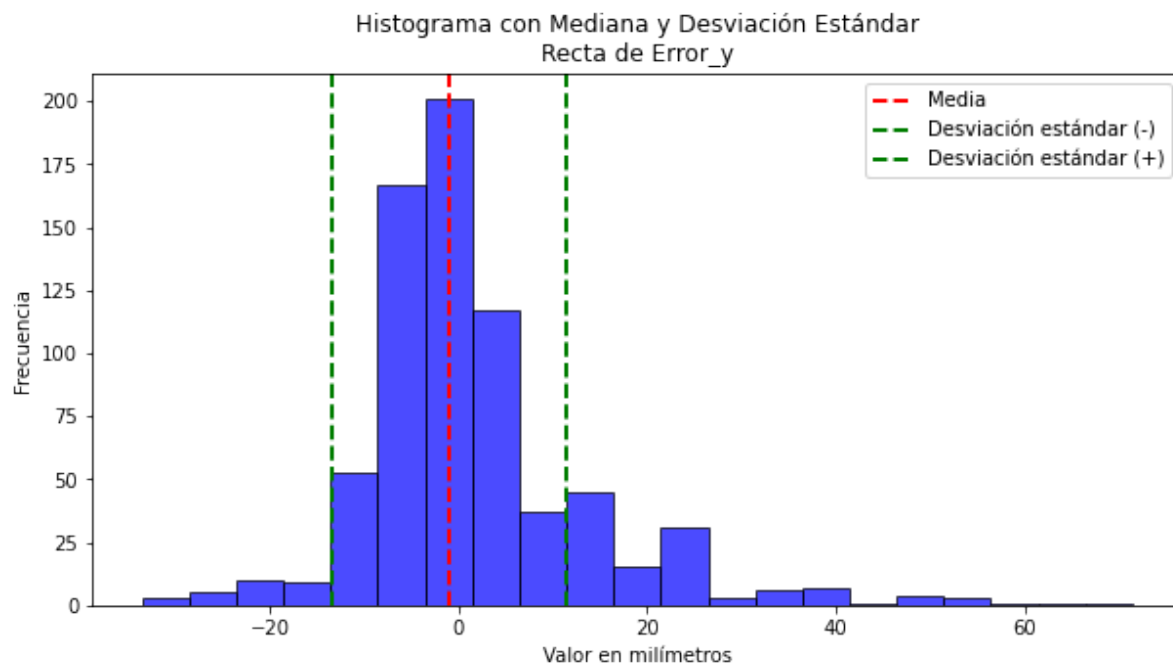
Mediana en Error_x: 0.311185

Desviación Estandar en Error_x: 18.59312008572265



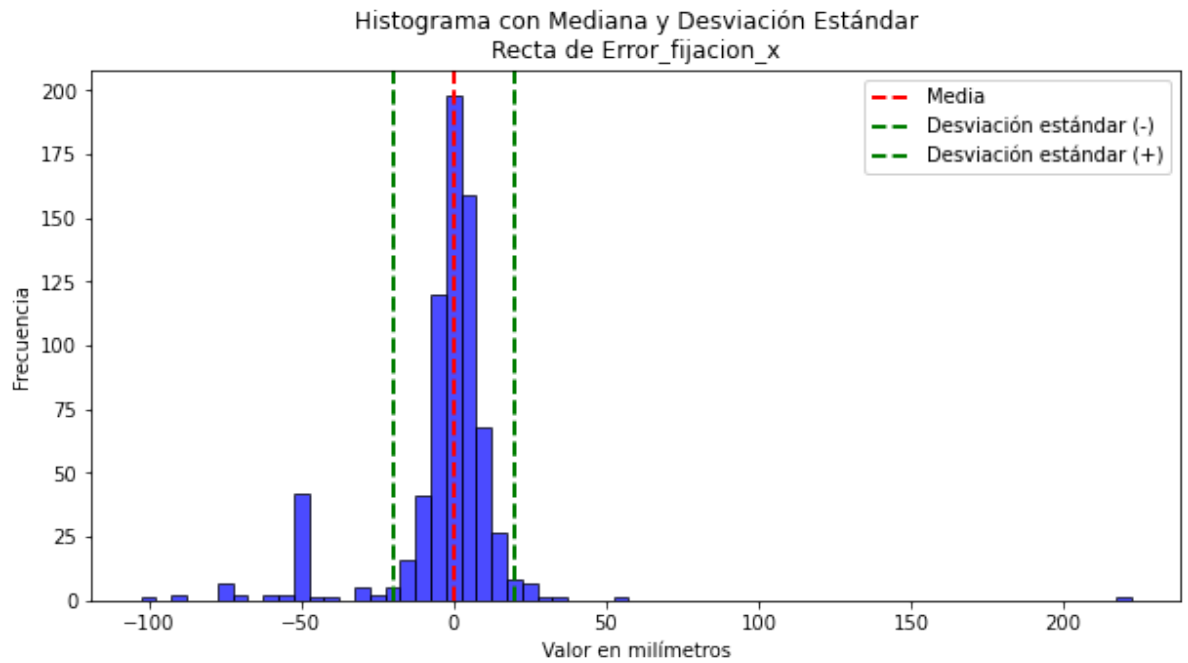
Mediana en Error_y: -0.956255

Desviación Estandar en Error_y: 12.420014663464226



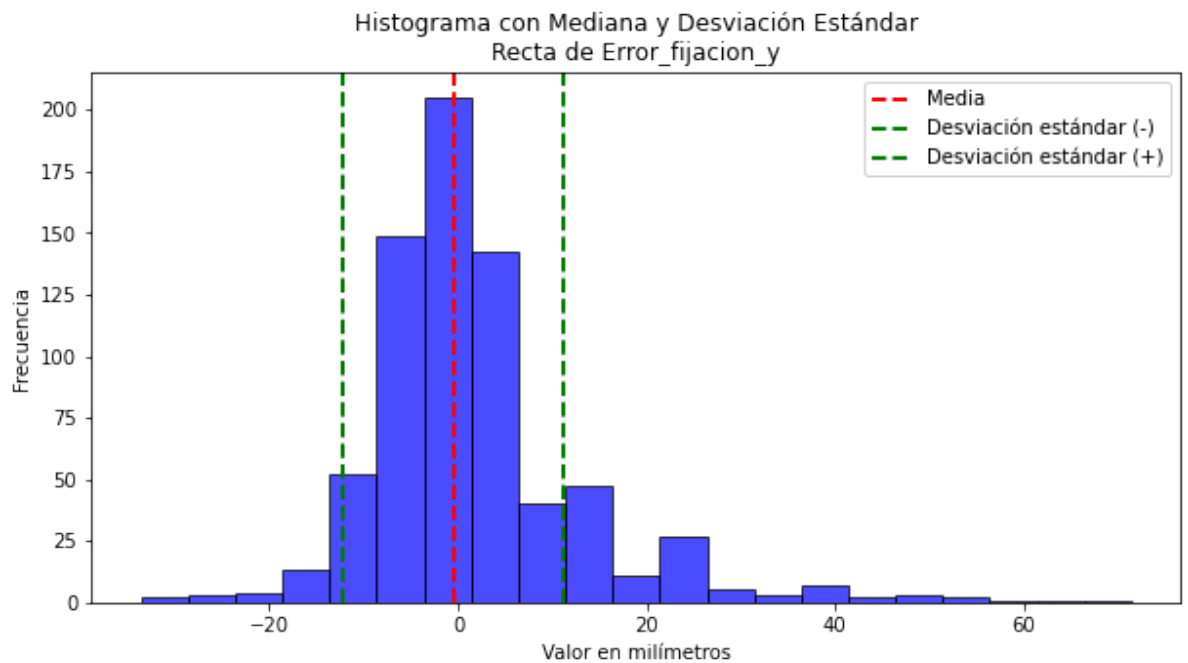
Mediana en Error_fijacion_x: 0.327115

Desviación Estandar en Error_fijacion_x: 19.846519548862453



Mediana en Error_fijacion_y: -0.49983500000000003

Desviación Estandar en Error_fijacion_y: 11.71941768949207



In []: