

```
# Función para crear un histograma general para cromosomas o plásmidos
def crear_histograma(resultados, titulo, etiqueta_x, etiqueta_y, archivo_salida_pdf):
    """
    Genera gráficos de barras para los porcentajes de GC y AT de cada cromosoma o
    plásmido y los guarda en un archivo PDF.
    """
    with PdfPages(archivo_salida_pdf) as pdf_pages:
        for secuencia in resultados:
            fig, ax = plt.subplots(figsize=(10, 6))

            porcentajes_gc = [region["GC%"] for region in secuencia["regiones"]]
            porcentajes_at = [region["AT%"] for region in secuencia["regiones"]]

            ancho_barras = 0.35 # Ancho de las barras
            indices_regiones = range(len(porcentajes_gc))
            ax.bar(indices_regiones, porcentajes_gc, ancho_barras, label='GC%',
color='green')
            ax.bar(
                [indice + ancho_barras for indice in indices_regiones],
                porcentajes_at,
                ancho_barras,
                label='AT%',
                color='blue',
            )

            ax.set_xlabel(etiqueta_x)
            ax.set_ylabel(etiqueta_y)
            ax.set_title(f'{titulo} - {secuencia["nombre"]}')
            ax.set_xticks([indice + ancho_barras / 2 for indice in indices_regiones])
            ax.set_xticklabels([]) # Eliminar etiquetas en el eje X
            ax.legend()

            pdf_pages.savefig(fig)
            plt.close(fig)
```

Esta función genera histogramas de los porcentajes de GC y AT para cada cromosoma o plásmido contenido en los resultados y los guarda en un archivo PDF. Veamos cómo funciona en detalle:

1. Definición de la función

```
def crear_histograma(resultados, titulo, etiqueta_x, etiqueta_y, archivo_salida_pdf):
```

- **resultados:** Lista con los análisis de secuencias, incluyendo nombres y regiones con porcentajes de GC y AT.
- **titulo:** Título general del gráfico.
- **etiqueta_x:** Etiqueta para el eje X (en este caso, indica las regiones o ventanas).
- **etiqueta_y:** Etiqueta para el eje Y (por ejemplo, "% de nucleótidos").
- **archivo_salida_pdf:** Nombre del archivo PDF donde se guardarán los gráficos.

2. Creación del archivo PDF

```
with PdfPages(archivo_salida_pdf) as pdf_pages:
```

- **PdfPages:** Clase de `matplotlib.backends.backend_pdf` que permite guardar múltiples gráficos en un archivo PDF.

3. Iteración por cada secuencia

```
for secuencia in resultados:
```

- Se recorre cada secuencia en `resultados` para generar un gráfico individual.

4. Extracción de datos

```
porcentajes_gc = [region["GC%"] for region in secuencia["regiones"]]  
porcentajes_at = [region["AT%"] for region in secuencia["regiones"]]
```

- Se crean listas que contienen los porcentajes de GC y AT de las regiones de la secuencia actual.

5. Configuración del gráfico

```
fig, ax = plt.subplots(figsize=(10, 6))
```

- Se crea una figura (`fig`) y un eje (`ax`) con un tamaño de 10x6 pulgadas.

6. Configuración de las barras

```
ancho_barras = 0.35 # Ancho de las barras  
indices_regiones = range(len(porcentajes_gc))
```

- **ancho_barras:** Define el grosor de las barras.
- **indices_regiones:** Crea una lista de índices para posicionar las barras en el gráfico.

```
ax.bar(indices_regiones, porcentajes_gc, ancho_barras, label='GC%', color='green')  
ax.bar(  
    [indice + ancho_barras for indice in indices_regiones],  
    porcentajes_at,  
    ancho_barras,  
    label='AT%',  
    color='blue',  
)
```

- Se crean dos conjuntos de barras:
 - Las barras para GC% comienzan en los índices originales (`indices_regiones`).
 - Las barras para AT% están desplazadas a la derecha por el ancho de las barras (`indice + ancho_barras`).

7. Etiquetas y formato

```
ax.set_xlabel(etiqueta_x)  
ax.set_ylabel(etiqueta_y)  
ax.set_title(f'{titulo} - {secuencia["nombre"]}')  
ax.set_xticks([indice + ancho_barras / 2 for indice in indices_regiones])  
ax.set_xticklabels([]) # Eliminar etiquetas en el eje X  
ax.legend()
```

- **ax.set_xlabel** y **ax.set_ylabel:** Configuran las etiquetas de los ejes.
- **ax.set_title:** Asigna un título al gráfico, que incluye el nombre de la secuencia.

- `ax.set_xticks`: Ajusta las posiciones de las marcas del eje X.
- `ax.set_xticklabels([])`: Elimina las etiquetas del eje X.
- `ax.legend`: Agrega una leyenda para las barras (GC% y AT%).

8. Guardado del gráfico

```
pdf_pages.savefig(fig)
plt.close(fig)
```

- `pdf_pages.savefig(fig)`: Guarda el gráfico actual en el archivo PDF.
- `plt.close(fig)`: Cierra la figura para liberar memoria.

Ejemplo de uso

```
resultados = [
    {
        "nombre": "Cromosoma_1",
        "regiones": [{"GC%": 40.0, "AT%": 60.0}, {"GC%": 50.0, "AT%": 50.0}],
    },
    {
        "nombre": "Cromosoma_2",
        "regiones": [{"GC%": 35.0, "AT%": 65.0}, {"GC%": 45.0, "AT%": 55.0}],
    },
]

crear_histograma(
    resultados,
    titulo="Porcentaje de GC y AT por región",
    etiqueta_x="Regiones",
    etiqueta_y="Porcentaje",
    archivo_salida_pdf="histogramas.pdf",
)
```

Esto generará un archivo PDF (histogramas.pdf) con gráficos de barras para cada cromosoma. Cada gráfico mostrará los porcentajes de GC y AT por región, en colores verde y azul respectivamente.