

```
# Función para guardar los resultados en un archivo de texto
def guardar_resultados_en_archivo(resultados, nombre_archivo_salida):
    """
    Escribe los resultados del análisis en un archivo de texto, incluyendo la asignación
    de bandas R o G.
    """
    with open(nombre_archivo_salida, "w") as archivo_salida:
        for secuencia in resultados:
            archivo_salida.write(f"{secuencia['nombre']}\n")
            conteo_bandas_r = 0
            conteo_bandas_g = 0

            for indice_region, region in enumerate(secuencia["regiones"], 1):
                gc = region["GC%"]
                at = region["AT%"]
                tipo_banda = "GC Band" if gc > at else "AT Band"
                banda = "G" if tipo_banda == "GC Band" else "R"
                archivo_salida.write(
                    f"Región {indice_region} - {tipo_banda}: GC% = {gc:.4f}, AT% = {at:.4f} ----> Banda: {banda}\n"
                )
                if banda == "G":
                    conteo_bandas_g += 1
                else:
                    conteo_bandas_r += 1

            archivo_salida.write(f"Resumen para {secuencia['nombre']}: \n")
            archivo_salida.write(f"Total de Bandas G: {conteo_bandas_g} \n")
            archivo_salida.write(f"Total de Bandas R: {conteo_bandas_r} \n")
```

Esta función toma los resultados del análisis de secuencias y los guarda en un archivo de texto. Adicionalmente, asigna bandas "G" o "R" según los porcentajes de GC y AT de cada región.

1. Definición de la función

```
def guardar_resultados_en_archivo(resultados, nombre_archivo_salida):
```

- **resultados:** Una lista de diccionarios, donde cada diccionario representa los datos analizados para una secuencia, incluyendo sus regiones y porcentajes GC/AT.
- **nombre_archivo_salida:** Nombre del archivo donde se guardará el texto con los resultados.

2. Apertura del archivo

```
with open(nombre_archivo_salida, "w") as archivo_salida:
```

- El archivo se abre en modo escritura ("w"). Si el archivo ya existe, será sobrescrito.
- **with:** Garantiza que el archivo se cierre automáticamente al finalizar la operación.

3. Iteración por secuencia

```
for secuencia in resultados:
    archivo_salida.write(f"{secuencia['nombre']}\n")
```

- **secuencia:** Cada elemento de la lista `resultados` representa una secuencia analizada.
- **secuencia['nombre']:** Nombre de la secuencia, que se escribe al inicio de los resultados correspondientes.

4. Inicialización de contadores

```
conteo_bandas_r = 0
conteo_bandas_g = 0
```

- Variables para contar las bandas asignadas como "R" (ricas en AT) y "G" (ricas en GC).

5. Procesamiento de regiones

```
for indice_region, region in enumerate(secuencia["regiones"], 1):
    gc = region["GC%"]
    at = region["AT%"]
```

- `secuencia["regiones"]`: Lista de regiones analizadas para esta secuencia.
- `indice_region`: Índice de la región, enumerado desde 1.
- Se extraen los valores de GC% y AT% de cada región.

6. Asignación de bandas

```
tipo_banda = "GC Band" if gc > at else "AT Band"
banda = "G" if tipo_banda == "GC Band" else "R"
```

- Si el porcentaje de GC es mayor que el de AT, se asigna la banda como "GC Band" y, por tanto, "G".
- Si AT es mayor, la banda se asigna como "AT Band" y "R".

7. Escritura de detalles de la región

```
archivo_salida.write(
    f" Región {indice_region} - {tipo_banda}: GC% = {gc:.4f}, AT% = {at:.4f} ---->
Banda: {banda}\n"
)
```

- Los datos de la región se escriben en el archivo:
 - Índice de la región.
 - Tipo de banda (GC Band o AT Band).
 - Porcentajes de GC y AT con 4 decimales.
 - Banda asignada (G o R).

8. Actualización de contadores

- ```
if banda == "G":
 conteo_bandas_g += 1
else:
 conteo_bandas_r += 1
```

- Se incrementa el contador correspondiente según la banda asignada.

## 9. Escritura del resumen por secuencia

```
archivo_salida.write(f" Resumen para {secuencia['nombre']}: \n")
archivo_salida.write(f" Total de Bandas G: {conteo_bandas_g} \n")
archivo_salida.write(f" Total de Bandas R: {conteo_bandas_r} \n")
```

- Después de procesar todas las regiones de una secuencia, se escribe un resumen indicando cuántas bandas G y R se encontraron.

## Ejemplo de uso

```
resultados = [
 {
 "nombre": "Secuencia_1",
 "regiones": [{"GC%": 55.0, "AT%": 45.0}, {"GC%": 40.0, "AT%": 60.0}],
 }
]
guardar_resultados_en_archivo(resultados, "resultados.txt")
```

El archivo resultados.txt podría contener:

```
Secuencia_1
 Región 1 - GC Band: GC% = 55.0000, AT% = 45.0000 ----> Banda: G
 Región 2 - AT Band: GC% = 40.0000, AT% = 60.0000 ----> Banda: R
Resumen para Secuencia_1:
 Total de Bandas G: 1
 Total de Bandas R: 1
```