Análisis formal de complejidad para `counting_sort_volumen`

Sea:

- n = número de elementos del arreglo `arr`
- k = valor máximo entero de `volumen_total * 100` (guardado como `max_val`)

1. Planteamiento y desarrollo de $T(n, k)$
Analicemos cada sección del algoritmo:

- Asignación y verificación inicial: t_0 (constante)
- Búsqueda de max_val: ciclo de $i = 1$ a $n - 1$ (t_1 por iteración)

$$T_1(n) = \sum_{i=1}^{n-1} t_1 = (n-1)t_1$$

- Inicialización del arreglo $count[0..k]$: ciclo de $i = 0$ a k (t_2 por iteración)

$$T_2(k) = \sum_{i=0}^{k} t_2 = (k+1)t_2$$

- Conteo de frecuencias: ciclo de $i = 0$ a $n - 1$ (t_3 por iteración)

$$T_3(n) = \sum_{i=0}^{n-1} t_3 = nt_3$$

- Suma acumulada de frecuencias: ciclo de $i = 1$ a k (t_4 por iteración)

$$T_4(k) = \sum_{i=1}^{k} t_4 = kt_4$$

- Reordenamiento: ciclo de $i = n - 1$ a 0 (t_5 por iteración)

$$T_5(n) = \sum_{i=0}^{n-1} t_5 = nt_5$$

- Copia del arreglo ordenado: ciclo de $i = 0$ a $n - 1$ (t_6 por iteración)

$$T_6(n) = \sum_{i=0}^{n-1} t_6 = nt_6$$

Sumando todos los términos:

$$T(n,k) = t_0 + (n-1)t_1 + (k+1)t_2 + nt_3 + kt_4 + nt_5 + nt_6$$

Agrupando términos:

$$T(n,k) = c_1 n + c_2 k + c_3$$

donde $c_1 = t_1 + t_3 + t_5 + t_6$, $c_2 = t_2 + t_4$, $c_3 = t_0 - t_1 + t_2$

$$T(n,k) = c_1 n + c_2 k + c_3$$

---

2. Calculando los casos

Mejor caso: Si $k \ll n$ (rango de valores pequeño), domina n:

$$T_m(n,k) \approx c_1 n + c_3$$

$$T_m(n,k) \in \Theta(n)$$

Peor caso: Si $k \gg n$ (rango de valores grande), domina k:

$$T_p(n,k) \approx c_2 k + c_3$$

$$T_p(n,k) \in \Theta(k)$$

Caso promedio: En general, el algoritmo tiene costo proporcional a $n+k$:

$$T_{pr}(n,k) = c_1 n + c_2 k + c_3$$

$$T_{pr}(n,k) \in \Theta(n+k)$$

---

3. Comprobación de cotas por límites

$$\lim_{n\to\infty} \frac{T(n,k)}{n} = c_1 \qquad \text{(si } k \text{ es fijo)}$$

$$\lim_{k\to\infty} \frac{T(n,k)}{k} = c_2 \qquad \text{(si } n \text{ es fijo)}$$

$$\lim_{n,k\to\infty} \frac{T(n,k)}{n+k} = \text{constante}$$

Por lo tanto:

$$T(n,k) \in \mathcal{O}(n+k)$$

$$T(n,k) \in \Omega(n+k)$$

$$T(n,k) \in \Theta(n+k)$$

4. Resumen Final:

- Mejor caso: $\Theta(n)$
- Peor caso: $\Theta(k)$
- Caso promedio: $\Theta(n + k)$

Cada cota ha sido verificada con límites y se han desarrollado las sumatorias con los nombres reales del código.