Sergi Mestres, Ferran Velasco, David Valero i Nil Vilas

3. El coeficient de Gini és una mesura de la desigualtat ideada per l'estadístic italià Corrado Gini. Normalment s'utilitza per mesurar la desigualtat en els ingressos, dins d'un país, però pot utilitzar-se per mesurar qualsevol forma de distribució desigual. El coeficient de Gini és un nombre entre 0 i 1, on 0 es correspon amb la perfecta igualtat (tots tenen els mateixos ingressos) i on el valor 1 es correspon amb la perfecta desigualtat (una persona té tots els ingressos i els altres cap).
Formalment, si $r = (r_1, \ldots, r_n)$, amb $n > 1$, és un vector de valors no negatius, el coeficient de Gini es defineix com:

$$G(r) = \frac{\sum_{i=1}^{n}\sum_{j=1}^{n}|r_i - r_j|}{2(n-1)\sum_{i=1}^{n} r_i},$$

Proporcioneu un algorisme eficient per calcular el coeficient de Gini donat el vector r

La aplicació directa de la fórmula dóna com a resultat un algorisme de cost $O(n^3)$, però si el vector d'entrada està ordenat, cost $O(n \log n)$, podem trobar la solució de manera lineal, cost $O(n)$, donant una cost combinat de $O(n \log n + n)$.

Per començar, cal analitzar els resultats del valor absolut. Donats dos valors r_i i r_j emparellats per primera vegada, en què $j = i + 1$ pensem:

- Aquests valors a la fórmula apareixeran aparellats dues vegades, i el valor absolut de totes dues operacions serà el mateix. Per tant, podem fer la operació dues vegades quan els aparellem per primera vegada

- Ja que els valors estan ordenats en el vector, $r_i < r_{j..n}$. Per aquest motiu, podem saber que $|r_i - r_j| = r_j - r_i$, i el mateix per tots els següents valors de j fins a n

- Donat l'anterior, podem saber que r_i apareixerà en la resta quan ens desfem del valor absolut tantes vegades com valors superiors a i, i serà el valor positiu tantes vegades com valors més petits de i hi hagi, tot això comptat dues vegades, ja que els aparellaments es produeixen en dues ocasions

-Ja que al denominador apareix un 2, podem deixar de comptar dues vegades els aparellaments i treure aquest 2 de la fórmula

Donat l'anterior, definim un algorisme que tingui en compte tot l'esmenat:

```
float coef_gini(vector<int> & v) {
	int n = v.size();
	int val, sum; val = sum = 0;
	for (int i = 0; i < n; ++i) {
		val = val + v[i]*(i – (n – i - 1));
		sum += v[i];
	}
	return float(val / (sum * (n – 1)));
}
```